Seguimiento visual de objetos articulados utilizando modelos gráficos basados en energía

María Alejandra Dávila Salazar

Departamento de Ingeniería Eléctrica y Electrónica
Universidad del Norte

8 de julio de 2014

Seguimiento visual de objetos articulados utilizando modelos gráficos basados en energía

María Alejandra Dávila Salazar

Tesis presentada como requisito parcial para optar el título de Magíster en Ingeniería Electrónica

Director:
Juan Carlos Niebles Duque

Departamento de Ingeniería Eléctrica y Electrónica
Universidad del Norte

8 de julio de 2014

A Dios,
a mi mamá, a mi papá
y a mis hermanas.

Índice

Lista de Figuras

Lista de Tablas

Agradecimientos

Este trabajo fue posible gracias a Dios, quien me ha dado la sabiduría, perseverancia y fortaleza para lograr con éxito mis metas.

Le agradezco a mi asesor, el profesor Juan Carlos Niebles, cuyo acompañamiento ha sido indispensable; y sus ideas, sus desarrollos, su apoyo y su estímulo se convirtieron en las principales herramientas para construir este trabajo. Gracias por compartir conmigo su conocimiento y experiencia.

A mi mamá, mi papá y mis hermanas, quienes son mi inspiración y motivación en la vida. Por la confianza que han puesto en mí, por el orgullo que sienten por mí, por cuidarme, por consentirme y brindarme tanto amor, por estar siempre a mi lado. Gracias, infinitas gracias. A ustedes les dedico este trabajo y esta nueva meta que he logrado en mi carrera profesional.

Quiero expresar unas palabras de agradecimiento a todos mis amigos y a mis compañeros del laboratorio de Ingeniería Eléctrica y Electrónica, que me brindaron su ayuda, motivación y su amistad, importante para culminar satisfactoriamente la maestría. Gracias por su compañía, por las discusiones trascendentales y por los momentos de esparcimiento que compartimos juntos.

De manera especial agradezco a Sulay y Manuel, quienes me han ofrecido su amistad sincera e incondicional a pesar de la distancia. Gracias a su apoyo, a sus consejos y regaños he tenido la motivación para continuar mi proceso de formación y esforzarme por cumplir los logros que me he propuesto.

Por último, agradezco a las entidades que me brindaron su apoyo económico durante esta etapa. A Colciencias y su programa de Joven Investigador. Al Ministerio de Tecnologías de la Información y las Comunicaciones (Mintic) y al Icetex, que me otorgaron la Beca para estudiar la Maestría por medio del Fondo Talento Digital.

Resumen

El seguimiento visual de objetos es un tema de interés en el campo de visión por computador, debido a que permite fortalecer las líneas de investigación de reconocimiento de acciones, resumen de video o detección de eventos. Además por su utilidad en aplicaciones como sistemas de monitoreo de personas, edificaciones o tráfico vehicular, que utilizan sistemas de visión, donde resulta importante un método de seguimiento automático para apoyar la tarea de vigilancia o supervisión en el cuidado de ancianos, niños o personas enfermas. Esta investigación esta centrada en el desarrollo de un sistema computacional capaz de localizar y seguir un objeto a través de una secuencia de video, utilizando un modelo de grafos basado en un modelo de energía. Este sistema está enfocado en seguir objetos relevantes presentes en el video, dándole prioridad a elementos sobresalientes con respecto al fondo, en movimiento, en primer plano, entre otras características predominantes. Para el sistema, un objeto está delimitado por un contorno sin forma fija, con el fin de adaptar la localización resultante a la forma precisa del objeto. El rendimiento de este modelo se ha verificado en tres conjuntos de videos Cars-MoSeg, SegTrack y un conjunto propio MCCD, obteniendo resultados comparables con investigaciones similares. El buen desempeño del algoritmo en la tarea de seguimiento se ha demostrado cualitativamente presentando algunas secuencias de imágenes.

Palabras Claves

Visión por computador, Seguimiento Visual, Objetos articulados, Métodos de aprendizaje de máquina, Segmentación, Modelos basados en Grafos.

Introducción

En visión por computador, la tarea de seguimiento de objetos es abordada por los investigadores para diferentes fines, como herramienta de caracterización en el reconocimiento de acciones o como punto central en aplicaciones de robótica. En los últimos años se han desarrollado modernas y atractivas tecnologías que ofrecen los nuevos dispositivos electrónicos, como el uso de videojuegos, computadores y televisores sin controles [1, 2], los cuales funcionan con las instrucciones captadas del movimiento de la persona. En el campo de visión por computador, se han desarrollado múltiples herramientas que soportan estos adelantos tecnológicos, como sistemas de reconocimiento de actividades [3], detección de objetos [4] y seguimiento [5].

Este último ha contribuido en diferentes aplicaciones, como en los sistemas de vigilancia para rastreo y búsqueda de intrusos [6, 7], siendo muy útil ya que puede cubrir todos los sucesos que una persona encargada de la vigilancia no puede detectar, por tener que revisar múltiples cámaras simultáneamente; como en el área de construcción, para monitoreo continuo de obras civiles [8], con el cual se puede velar por el bienestar de los obreros, el correcto desempeño de sus funciones y controlar el avance de la obra; o en monitoreo de tráfico vehicular [9, 10]. Además, el seguimiento de objetos, permite el desarrollo de tecnologías en el campo de robótica, en actividades de interacción hombre-máquina [11], donde se utiliza el movimiento de la mano como herramienta para mover el puntero del computador; en juegos por control de movimiento [12, 13], como el Xbox con sensor Kinect® [14]; o en torneos deportivos como el Grand Slam en tenis donde se implementan sistemas [15] para verificación de puntos y faltas, en los que se requiere seguimiento de la pelota o de los movimientos del jugador.

El seguimiento visual es una técnica que implica la identificación de personas,

animales u objetos en una imagen y permite su continua localización a través del tiempo en una secuencia de video. Esta herramienta resulta muy útil y necesaria para el análisis de información a partir de imágenes, como en reconocimiento de actividades [16], detección de eventos, resumen de video, búsqueda de videos basada en contenido, donde se desea extraer características del comportamiento de una persona o un objeto, pero se requiere para ello conocer su ubicación en cada instante de tiempo, por lo cual resulta importante para el área de visión por computador dado que fortalece y complementa el rendimiento de dichas técnicas. A pesar que este tema se ha abordado desde múltiples enfoques, como los trabajos en [6, 17, 18], se requiere que las técnicas desarrolladas ofrezcan resultados óptimos y confiables, con bajo costo y complejidad computacional. El trabajo de [19], plantea que ningún modelo propuesto está exento de errores, sin embargo estos se pueden reducir si se tienen en cuenta los errores más frecuentes y se intenta corregirlos. El trabajo en [4], propone un sistema que es capaz de detectar algunos errores y aprender de ellos para optimizar el resultado. Aunque no puede evitar todos los errores posibles, es capaz de reducirlos considerando su existencia en las nuevas secuencias de imágenes analizadas.

Las aplicaciones de seguimiento de objetos, tales como los sistemas de vigilancia, exigen resultados más precisos y mayor independencia del usuario. Bajo estas necesidades se han planteado en este campo tres retos para investigar: La inicialización automática, el seguimiento de múltiples objetos y la localización adaptada a objetos articulados. El primer reto se refiere a la necesidad de un sistema de seguimiento que disminuya la supervisión continua del usuario, en cuanto no se solicite la ubicación inicial del objeto, con el fin de aumentar la eficiencia y productividad, por ejemplo, en una tarea de vigilancia. El segundo reto se asocia a la posibilidad de detectar más de un objeto simultáneamente; considerando que es lo más común encontrar varias personas, animales y objetos en una escena o en una misma secuencia de video. El tercer reto se enfoca en mejorar la precisión en la localización del objeto, esto se puede conseguir si se define un contorno adaptado a la forma del objeto de interés en lugar de rectángulos o elipses de formas y proporciones fijas.

Esta investigación se enfoca en el problema de seguir objetos articulados, esto es lograr la localización y seguimiento de objetos adaptándose a su contorno. Para

esto se diseña un sistema con una técnica de seguimiento eficiente, fundamentada en un modelo de grafos basado en energía, capaz de adquirir información del objeto de interés en cada *frame* y tomar decisiones a partir de cada imagen que va recibiendo, sin requerir la secuencia de video completa para su procesamiento. Además, se define con precisión el objeto de interés por medio de un contorno adaptado a la forma que tome éste en cada instante.

Objetivos

Objetivo General

OG1: Diseñar e implementar un sistema de seguimiento de objetos en secuencias de imágenes con aprendizaje de máquina.

Objetivos Específicos

OE1: Consultar las técnicas y metodologías utilizadas en el campo de visión por computador para seguimiento y aprendizaje de máquina.
OE2: Recolectar un conjunto de videos de contextos diferentes, para ser utilizados en el análisis, desarrollo y evaluación del sistema.
OE3: Diseñar e implementar un algoritmo que sea capaz de controlar el seguimiento de elementos en una imagen a través de la secuencia de un video, basado en métodos de aprendizaje de máquina.
OE4: Verificar experimentalmente el funcionamiento del sistema y cuantificar su rendimiento.

Contribución

La principal contribución de esta investigación está en el desarrollo de una técnica capaz de seguir objetos independientemente de la categoría a la que pertenezca. En esta propuesta la localización del objeto de interés no se define por elipses o rectángulos rígidos, sino por un contorno variable adaptado a la forma precisa del objeto. Además, se incluyen técnicas de aprendizaje de máquina en su diseño, como

un modelo de optimización basado en grafos para el etiquetado de la región que corresponde al objeto, así como la combinación de descriptores basados en puntos con descriptores de apariencia y movimiento basados en superpíxeles.

Como aporte al área de investigación, se recolecta y organiza un conjunto de videos de maquinarias en obas de construcción civil. Este conjunto se ha denominado MCCD y consiste en 10 secuencias cortas, grabadas a cielo abierto.

Resumen del Documento El resto del documento se organiza como se describe a continuación. En el Capítulo 1 se presenta la revisión del trabajo relacionado a esta investigación; En el Capítulo 2 se estudian y analizan dos trabajos relacionados, a partir de implementaciones propias y algunas modificaciones realizadas para acercarlo más al enfoque de este trabajo. En el Capítulo 3, se describe el diseño propuesto y cada uno de los detalles del algoritmo implementado. La descripción del conjunto de videos utilizados y los experimentos y resultados obtenidos, se describen en el Capítulo 4. Y finalmente, se presentan las Conclusiones de la investigación realizada.

Capítulo 1

Trabajos Relacionados

En visión por computador se desarrolla la idea de construir un sistema de cómputo capaz de simular la visión humana, pero más allá de mostrarle imágenes al computador, se pretende desarrollar procesos que permitan tomar decisiones basada en imágenes o videos, [20]. Una de las tareas que se abordan en este campo es el seguimiento visual, que consiste en la localización de un elemento en espacio y tiempo en una secuencia de video.

1.1. Seguimiento de Objetos

El seguimiento visual de objetos es una tarea de interés en el campo de visión por computador, útil en aplicaciones de vigilancia y monitoreo, control de tráfico vehicular, videojuegos, entre otras. Esta labor implica mantener la localización espacial y temporal de uno o varios objetos en una secuencia de video, como en [21]. Los investigadores han abordado esta tarea desde diferentes enfoques [6, 17, 18], como seguimiento por detección frame a frame, seguimiento de puntos de interés basado en trayectorias, utilizando métodos de aprendizaje a partir de modelos de objetos, con estructuras pictóricas, métodos de *level set* o como una técnica de segmentación.

Los métodos basados en trayectorias implican el seguimiento de puntos de interés pertenecientes al objeto a través de cada *frame* de la secuencia de video, como [21–23]. Aunque algunos trabajos realizan muestreo denso de puntos en un determinado frame, [24], la mayoría se enfoca en detectar puntos de esquinas o de bordes para

realizar el seguimiento. Cada punto es seguido de acuerdo a técnicas de estimación de movimiento, como flujo óptico (*optical flow*), que consiste en un vector de orientación indicando la ubicación más probable de un punto en el siguiente instante de tiempo. A partir de esta estimación se construye una trayectoria de cada punto que permite seguir la región de interés. Una alternativa utilizada en estimación de movimiento es el filtro Kalman, el cual ofrece un buen resultado, desde que predice el valor de un parámetro y en un proceso de optimización corrige el error en la predicción iterativamente.

La detección frame a frame, como método para seguimiento visual, consiste en analizar la secuencia de imágenes independientemente y definir la ubicación del objeto de interés en cada *frame*. Una ventaja de este método es que se puede detectar el objeto de interés en cualquier instante del video, a pesar que se haya perdido por oclusión. Por esto, la tarea de detección de objetos se ha convertido en un punto de interés para los investigadores de visión por computador interesados en seguimiento visual.

En la detección de objetos se usan modelos basados en partes o ventanas deslizantes. Un modelo basado en partes, pretende detectar un objeto definiéndolo como un conjunto de partes más simples, de tal forma que la detección busca las partes del objeto independientemente. La técnica de ventanas deslizantes, explora toda la imagen desplazándose de región en región (ventanas) hasta encontrar el objeto indicado en el cuadro delimitador inicial, [25]. Este último método implica escanear miles de opciones en el área de la imagen analizada, como [26] que se centra en la exploración de la imagen entera para encontrar objetos en función de características específicas, tales como el color, bordes y tamaño.

Una alternativa diferente que pretende optimizar la exploración de las regiones donde posiblemente exista un objeto, es la búsqueda selectiva, como [16, 27, 28]. Esta se usa para dar prioridad a determinadas zonas de una imagen, de modo que el algoritmo de detección sea implementado solamente para las áreas representativas y así se pueda reducir el costo computacional del sistema. La selección de las regiones más representativas puede ser de manera aleatoria o basada en características de prominencia. En [28] se plantea ésta técnica para reducir el costo computacional desperdiciado en explorar exhaustivamente las posibles ventanas donde exista un

objeto. Esta técnica pretende seleccionar ventanas candidatas, a partir de la segmentación de la imagen, para reconocer objetos independientemente de la clase a la que pertenezcan. Para esto hace uso de la segmentación jerárquica de [29] que consiste en una sobre-segmentación inicial, en el cual las regiones vecinas con apariencia similar (textura y color) se unen y crean las siguientes segmentaciones, este proceso se realiza iterativamente hasta que la región se convierte en la imagen completa. Las ventanas candidatas corresponden a cada segmento creado o al cuadro delimitador que mejor la ajuste.

El uso de estos métodos requiere determinar si existe un objeto o no, para esto se usa un modelo para representar cada categoría de objeto entrenada en el sistema, esto es un modelo para gato, perro, carro, silla, etc. Como en [30, 31], que se enfocan en el seguimiento de personas. En general, estos métodos se conocen como detección por categoría. Sin embargo, debido a la diversidad intra-clase, como la existencia de diferentes razas de perros, al punto de vista del objeto o a las condiciones de iluminación, resulta muy complejo aprender un modelo que reúna todas las posibles características y variaciones posibles. Por lo que se tiende a limitar estas técnicas para aplicaciones específicas, con muchas restricciones. Para sobrellevar esto, se pretende generalizar la detección caracterizando objetos, independiente de la clase.

Los métodos de detección independiente de la clase, como el tema de esta investigación y el trabajo de [16, 27], permiten abordar el problema de manera general sin desarrollar una técnica funcional para una sola aplicación específica. Esto permite ser aplicado en cualquier contexto, no requiere un extenso conjunto de datos para una posible etapa de entrenamiento, además soporta adaptabilidad a cualquier contexto nuevo, descubriendo objetos no vistos anteriormente. Métodos como sustracción de fondo, aplicado en secuencias de videos simples, son capaces de aislar un elemento de interés del resto de la escena, entregando un objeto sin discriminar la categoría a la que pertenece. Esta técnica consiste en suprimir las regiones de la secuencia de imágenes que permanezcan homogéneas, invariantes o estáticas, considerándose como escena para un video, de tal forma que la sección sobresaliente que quede se considera como el objeto de interés.

La tarea de seguimiento se puede desarrollar con métodos supervisados, no supervisados o semi-supervisados, esto se refiere a la información que debe proveer el

usuario para el funcionamiento del sistema. En particular, para el seguimiento de objetos la supervisión puede estar relacionada con la indicación del objeto en algún instante de la secuencia de video, precisando la posición con un cuadro delimitador. En trabajos relacionados se realiza detección supervisada utilizando un cuadro delimitador del objeto al inicio del video, como [32, 33]. Un método semi-supervisado, puede requerir que en cada video se entregue una etiqueta que indica si existe o no un objeto, sin especificar la ubicación. En la búsqueda de sistemas no supervisados, algunos investigadores [24] realizan la detección sin ninguna ubicación o etiqueta inicial de la posición o existencia del objeto.

Dependiendo de la representación del objeto de interés existen diferentes enfoques desde los cuales se puede desarrollar la tarea de seguimiento. Muchos investigadores han considerado que el objeto corresponde a una región delimitada por una forma geométrica fija, como una elipse o un rectángulo, lo que se conoce como delimitador o bounding box. También se considera una región definida por un delimitador ajustada al contorno exacto del objeto, este enfoque permite lograr mayor precisión en la ubicación del objeto, por lo cual se puede aplicar la tarea de seguimiento en objetos deformables o articulados [28]. Cuando se tiene una región, ya sea formada por un conjunto de puntos densos o disperso, definida por un contorno o una forma geométrica, se requiere describirlo por medio de un conjunto de características, asociadas a color, textura, apariencia, movimiento, entre otras. En algunos trabajos se utilizan, HOG, SIFT, HOF, entre otras.

Considerando que la mayoría de objetos son deformables, que al cambiar de posición o punto de vista, se muestra una forma diferente. Se han desarrollado investigaciones relacionadas con seguimiento de objetos articulados, como en [28, 34], que usan métodos que pretenden localizar el objeto y definir la forma precisa de dicho objeto en cada instante del video. Esto se ha realizado creando modelos del objeto, asociadas a un método de aprendizaje, como estructuras pictoriales. Otro enfoque, es el de segmentar la secuencia de video y realizar el seguimiento sobre las regiones resultantes, seleccionando aquellos segmentos que pertenezcan a un objeto.

1.2. Segmentación y Agrupamiento

Un enfoque para desarrollar la tarea de seguimiento delimitando el contorno del objeto, es utilizar técnicas de segmentación para crear zonas consistentes con similitud de apariencia o movimiento, y propagar su localización a través de la secuencia, como [26, 35, 36]. Esto es importante ya que si se usa un rectángulo delimitador para indicar la posición del objeto de interés, no se tiene precisión en el resultado cuando el objeto no es rígido sino articulado y deformable, como personas desplazándose, saltando o bailando. Mientras que la segmentación permite localizar con precisión un objeto con una forma geométrica no definida e incluso deformable en el tiempo.

La segmentación se puede realizar basada en características espaciales que resaltan un objeto del fondo, como el color, tamaño, forma, bordes, prominencia y otros. Además, las características de movimiento como el flujo óptico, también se pueden combinar con las características espaciales para detectar un objeto. También se puede realizar por métodos como segmentación basada en grafos o por segmentación aglomerativa, que se explican en [20].

Algunos investigadores de visión por computador han desarrollado técnicas para realizar la segmentación en imagen estáticas, como [16, 28] quienes agrupan regiones de una imagen, llamándolas superpíxeles, teniendo en cuenta las características comunes como color, textura, entre otros. Sin embargo, la segmentación de objetos en video ha abarcado gran interés en este campo de investigación, como [37, 38].

En trabajos como en [34] se ha extendido la segmentación a video, incluyendo información temporal, para esto crean un agrupamiento de las regiones de una imagen y regiones similares entre frames con respecto a características temporales; estos segmentos formados se conocen como supervoxels. Esta investigación se basa en la definición de las regiones propuestas de [26], para la detección de objetos con segmentación, considerando un enfoque similar a la sustracción de fondo. Además, se incluyen los modelos mixtos Gaussianos (GMM) en sus representaciones. Este modelo requiere un clasificador para definir los segmentos espacio-temporales que corresponden a un objeto. Para esta tarea se han implementado clasificadores Bayesianos, basado en temas probabilísticos, y máquinas de soporte vectorial (SVM), que es un clasificador discriminante que requiere datos de entrenamiento para aprender a tomar la decisión correcta. Los modelos basados en energía, han ocupado un lugar

importante para la ejecución de esta tarea de clasificación.

1.3. Modelos basados en Grafos

En el campo de aprendizaje de máquina se utilizan frecuentemente los modelos basados en energía (EBM), para solucionar problemas de aprendizaje probabilísticos o no-probabilísticos. Como lo expresa LeCun en [39], casi todos aquellos problemas de aprendizaje que se puedan plantear como un modelo gráfico o algún otro modelo estructurado, pueden ser resueltos utilizando un modelo basado en energía. Este tipo de modelos, captura las dependencias entre variables asociándole una medida de compatibilidad (energía) a cada una.

De acuerdo a [39, 40], las diferentes aplicaciones del aprendizaje de máquina se pueden categorizar en tres tareas principales, la de predicción, clasificación y toma de decisiones. En este enfoque se puede considerar como predicción, entre otras, aquellas tareas que indican si en determinada imagen o región se encuentra un objeto. De esta forma, se puede plantear un modelo de energía para abordar esta tarea, tal como lo ha hace Fulkerson en su trabajo [41], al unir segmentos por medio de este enfoque. Algunos de los trabajos relacionados plantean el problema por medio de modelos gráficos como [42, 43], los cuales se pueden resolver como un EBM.

1.4. Métricas de Desempeño

En las conferencias de los últimos meses se han encontrado trabajos más recientes, como el presentado en [44], en el cual se pretende organizar la evaluación de los algoritmos propuestos en el campo de seguimiento, por lo cual se ofrece un conjunto de códigos e implementaciones de diferentes algoritmos de seguimiento con diferentes enfoques, así como incluye un conjunto amplio de videos con su correspondiente anotación manual o *groundtruth*, definidos como un rectángulo alrededor del objeto.

La tarea de seguimiento se considera eficiente cuando se mantiene la localización del objeto de interés con buena precisión. Para evaluar este rendimiento se utiliza una medida del traslape entre la región encontrada por el sistema implementado y el delimitador real establecido por el usuario. Esta métrica ofrece una relación entre

el área de intersección y el área de unión de la región anotada manualmente como objeto y la salida de un algoritmo de localización. Otra métrica usada comúnmente por los investigadores, como en [34], corresponde al promedio de píxeles etiquetados erróneamente. Esto es, el número de píxeles que según el algoritmo pertenecen al objeto pero no lo son, junto con los píxeles que se indican como fondo pero no son; este valor se promedia para todos los *frames* de la secuencia.

A partir de estas métricas se puede medir el rendimiento del sistema, con una precisión a nivel de píxel en la salida. Además, se puede realizar comparaciones equivalentes con los trabajos de otros investigadores, que utilicen el mismo conjunto de datos de prueba.

Capítulo 2

Revisión de algoritmos previos

En el proceso de investigación, se usaron algunas técnicas de seguimiento analizadas en el estado del arte, como líneas de desarrollo base para el novedoso método propuesto. Para esto se implementaron y desarrollaron modificaciones en las técnicas estudiadas, además se verificó el rendimiento de cada algoritmo en un conjunto de videos. La realización de esta revisión, pretende analizar las ventajas de sus modelos, explorar diferentes técnicas, reconocer métodos eficientes, plantear mejoras para el perfeccionamiento de éstas, para su posible aplicación en nuestro sistema. Y así construir un modelo fuerte, robusto y bien consolidado cuyos fundamentos sean las mejores técnicas encontradas.

2.1. *Objectness*

En el desarrollo de la investigación se ha implementado una técnica de *objectness*, basado en el trabajo de [16]. El autor propone detectar objetos en imágenes, independientes de la categoría, utilizando cuadros delimitadores o ventanas para indicar la posición del objeto. La selección de estas regiones se realiza basada en una medida de prominencia. Este parámetro le da mayor valor a las regiones de la imagen que resalten más con respecto a las demás, en términos de color, textura o apariencia. Inicialmente se tiene un conjunto de 100.000 ventanas aleatorias uniformemente distribuidas en 4 escalas diferentes. Los autores en [16] definen una medida de *objectness*, que asigna un puntaje a una región de acuerdo a la posibilidad de

existencia de un objeto. Este método se basa en cuatro características para tomar las decisiones: prominencia de múltiples escalas, contraste de color, la densidad de borde, característica basada en superpíxel. La detección de objetos se basa en un clasificador de Bayes básico usando una o más características comunes.

La implementación y modificación de esta técnica consiste en realizar una segmentación espacio-temporal de cada frame de la secuencia, para obtener regiones con contornos no-geométricos más cercanos a la forma real de los objetos; luego, para cada segmento determinar una medida de *objectness*; finalmente, utilizar un clasificador para indicar cuales segmentos corresponden a un objeto y cuáles no, de acuerdo a las medidas calculadas. El clasificador implementado en este trabajo es una máquina de soporte vectorial SVM, la cual depende de un modelo aprendido en un entrenamiento previo. La diferencia con el trabajo de [16] consiste en aplicar la detección sobre toda una secuencia de video, en lugar de utilizarla para una única imagen; además, se toman segmentos en vez de rectángulos para explorar cada *frame* en busca de la ubicación del objeto.

En la Figura 2.1 se muestran un par de *frames* de un video de prueba, en este caso se nota que el brazo de la grúa que inicialmente no se observaba aparece en los siguientes *frames*. Sin embargo, es posible identificar la nueva región del brazo de la grúa como objeto, considerando sus características de apariencia y de movimiento similares a las otras partes del objeto a seguir.

Figura 2.1: Imágenes de una secuencia de maquinaria de construcción

En la Figura 2.2 se muestra el resultado de la segmentación, donde se notan múltiples segmentos, que de acuerdo a las medidas calculadas se pueden asociar como objeto; además, se tiene la salida del algoritmo en la que se muestran las

posibles regiones donde puede localizarse el objeto, definidas como las regiones con más alto valor de *objectness*.

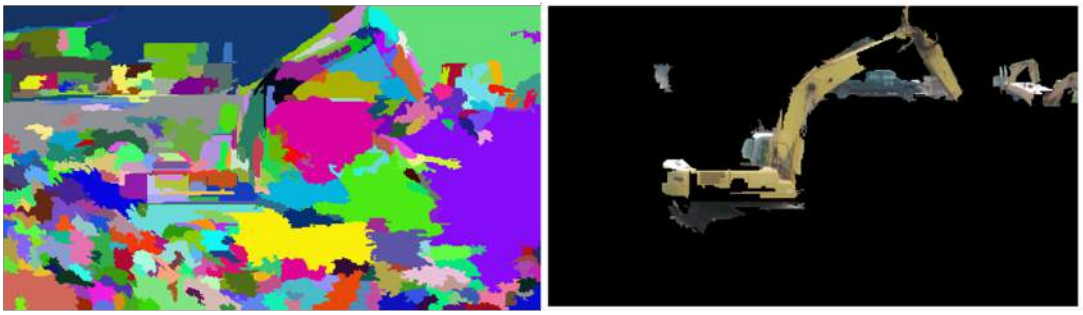

Figura 2.2: Implementación técnica de *objectness*. En la izquierda, resultado de la segmentación; en la derecha, regiones con alto valor de *objectness*

La implementación del clasificador se basa en una máquina de soporte vectorial (SVM) con kernel RBF (Radial Basis Function), para el cual se obtuvo el rendimiento mostrado en la Figura 2.3. En donde la matriz de confusión de la izquierda corresponde a la etapa de entrenamiento y la de la derecha a prueba. En este caso se observa que el clasificador presenta una tendencia a designar la mayoría de las regiones como fondo.

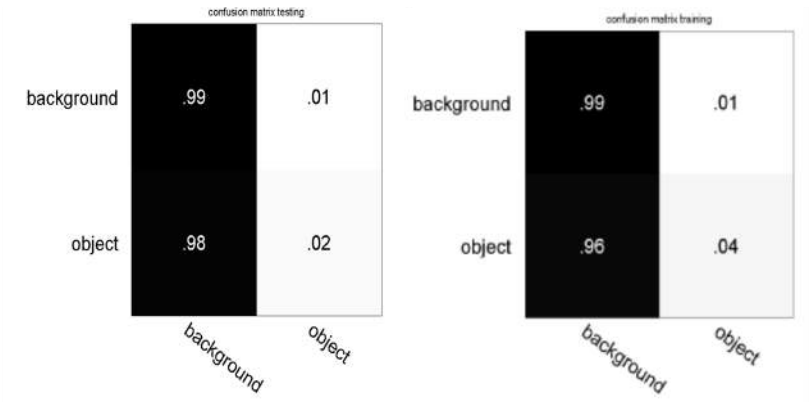

Figura 2.3: Matrices de confusión para el clasificador SVM. (Izquierda) etapa de entrenamiento, (Derecha) etapa de prueba

Conclusión A partir del estudio realizado se puede decir que el método de *objectness* presenta resultados aceptables. Sin embargo, su implementación en video y utilizando segmentos en lugar de rectángulos, no localiza correctamente el objeto. La principal causa de esto se asocia al entrenamiento previo, que se requiere para definir si un segmento es o no objeto; este entrenamiento necesita un conjunto de datos más grande, lo cual exige mayor tiempo de cómputo. Además, el uso de un entrenamiento como tal implica que el método para localizar objetos independientes de la categoría, requiera aprender un modelo del objeto para cada nuevo conjunto de videos.

2.2. Seguimiento Adaptativo

Revisando el trabajo relacionado en el tema de seguimiento de objetos, se tiene que la investigación de Stalder en [45], plantea una metodología cercana al trabajo que se espera desarrollar en este proyecto. Por esto, se realiza una implementación de esta técnica considerando las características y parámetros definidos en su trabajo. Este consiste en una etapa de detección de puntos de interés, predicción de posición basada en flujo óptico, creación de un modelo de objeto con descriptores SIFT para validar que la región de los puntos detectados correspondan a las características de un objeto (medida de objectness de [16]), segmentación por movimiento para detectar un nuevo objeto o actualizar el modelo del objeto según los grupos creados. Esto hace al seguimiento robusto a oclusiones, cambios abruptos de movimiento y a diferentes puntos de vista. Los resultados de esta implementación se muestran en la Figura 2.4, para un video de prueba utilizado por los autores en [45].

Para el proceso de seguimiento se ha desarrollado una técnica basada en flujo óptico (optical flow). Esta característica describe el vector de desplazamiento más probable que puede presentar cada punto de un frame en el siguiente instante de tiempo. Esto se realiza con la comparación de dos frames consecutivos del video. Para una optimización del algoritmo, el seguimiento se realiza sólo para ciertos puntos de interés detectados inicialmente, en lugar de hacerlo para cada uno de los píxeles como lo hace el autor; esto disminuye considerablemente el costo computacional de esta técnica. Estos puntos de interés corresponden a puntos de esquina encontrados

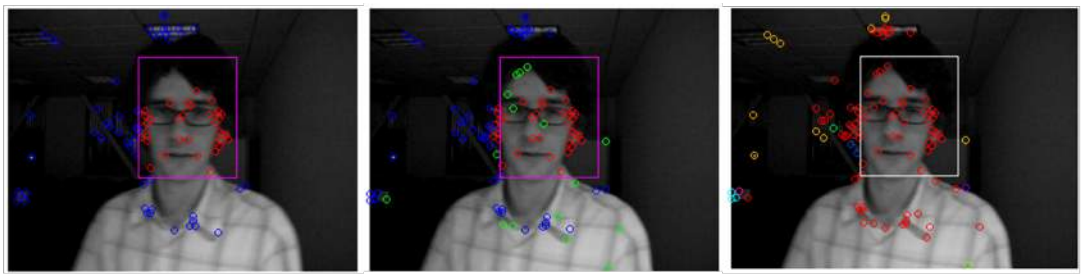

Figura 2.4: Resultados del seguimiento adaptativo en un video de prueba. En la izquierda, detección de puntos iniciales con la anotación manual. En el centro, la propagación y clasificación de puntos según el modelo. En la derecha, puntos con color de acuerdo al agrupamiento y salida del algoritmo indicada con un rectángulo en blanco.

con el detector de esquinas de Harris. En la imagen de la izquierda de la Figura 2.4, se muestra la inicialización manual del sistema, donde los puntos en rojo dentro del rectángulo son etiquetados como puntos de 'objeto', mientras que los puntos azules por fuera son asignados como puntos de 'escena' o fondo. La imagen del centro, contiene puntos en verde, que son anotados como 'No Vistos', los cuales corresponden a nuevos puntos detectados en los frames posteriores.

Para cada uno de los puntos de objeto detectados, se computa un descriptor SIFT con el cual se construye un modelo del objeto. Luego de tener el seguimiento de los puntos pertenecientes al objeto, se realiza una actualización del objeto. Para ello se utilizan métodos de agrupamiento de puntos basado en características de posición y movimiento, tomado de [29, 46], de tal forma que se obtengan regiones con similar movimiento, que posteriormente pueden ser identificadas como un objeto. En esta implmentación, se avaluaron tres métodos de agrupamiento diferentes,*Mean Shift*, Basado en Grafos y GANC (*Greedy Aglomerative Normalized Cut*).

A continuación se calculan dos medidas de objectness a partir de los grupos creados por los puntos de interés, en conjunto con nuevas regiones rectangulares que se crean alrededor de la posición anterior del objeto. Con esto se tiene un parámetro que mide la posibilidad de encontrar un objeto en cierta región. La medida de *Target Score* pretende cuantificar el porcentaje de puntos etiquetados como objeto que pertenecen a un grupo de puntos. En la Ecuación 2.1 se muestra la correspondiente expresión matemática.

$$TargetScore(s) = \frac{Num_{obj}(s) - Num_{esc}(s)}{Num(s)} \tag{2.1}$$

La segunda medida es *Dynamic Objectness* y mide la cantidad de puntos de un grupo que caen dentro de cada región particular, pero favoreciendo aquellos grupos que tienen la mayoría de puntos por fuera o la mayoría de puntos por fuera de la región definida. En la Ecuación 2.2, se observa la expresión correspondiente, en la cual c se refiere a los grupos formados por el agrupamiento de puntos y s a los segmentos de la imagen.

$$DynamicObj(s) = \sum_{c \in C} \frac{min(Num_{in}(s,c) - Num_{out}(s,c))}{Num(s)} \tag{2.2}$$

Finalmente, la selección de la región con las medidas más altas, entrega la posición del objeto en cada frame. Todo el procesamiento descrito, se realiza frame a frame, propagando la información con el modelo de SIFT construido y el seguimiento de los puntos por flujo óptico.

Revisando los valores de las métricas obtenidas se tiene que la medida de *Dynamic Objectness* no discrimina correctamente los grupos de puntos, por lo cual para esta implementación no es significativa para la selección de la región que contiene el objeto. Considerando que la medida que denominan como *Objectness* dinámico no captura estrictamente la pertenencia de un grupo de puntos dentro de una región rectangular, como era la intención; ya que esto falla en casos donde los grupos son muy pequeños con respecto a las regiones analizadas, entonces el valor tiende a ser cero, para todas las regiones, porque contiene todos los puntos de varios grupos. Mientras que la medida de *Target Score* que indica el porcentaje de puntos etiquetados como objeto en cada región evaluada, ofrece una contribución a la distinción de aquellas regiones que pueden contener un objeto, por lo que resulta relevante para localizar y mantener el seguimiento del objeto de interés.

Para evaluación del desempeño del sistema se utiliza una métrica conocida como *overlap* o solapamiento, la cual corresponde al porcentaje de solapamiento entre la región definida como real, anotada manualmente por el usuario, y la región entregada por el sistema implementado. Para la validación de esta implementación se utilizó un conjunto de videos de benchmark, utilizado en la investigación de [45], que están dis-

ponibles en [47]; esto con el fin de realizar una comparación adecuada. La Tabla 2.1 contiene los resultados promedio de los primeros frames, para los tres métodos de agrupamiento utilizados. Para *Mean Shift* el resultado promedio es 0.66, para el método basado en grafos 0.70 y para el método aglomerativo (GANC) 0.69. Es necesario mencionar, que este valor se calcula para los primeros 20 frames, considerando que después se pierde el seguimiento del objeto de interés; sin embargo, lo que se quiere resaltar en estos resultados es el desempeño de cada método de agrupamiento con respecto a los demás.

Tabla 2.1: Resultados de solapamiento para los primeros frames en los videos de prueba

Método de agrupamiento	**David**	**Dollar**	**Board**	**FaceOcc**
Mean Shift	0.32	0.92	0.65	0.76
Basado en grafos	0.34	0.92	0.76	0.77
GANC	0.35	0.92	0.70	0.77

Analizando los resultados obtenidos, se muestra una mínima diferencia en los resultados cuantitativos; sin embargo, para seleccionar el mejor método es necesario tener en cuenta otros factores. Por otro lado, el método de grafos y GANC utilizan una matriz de afinidad para relacionar la información de posición y movimiento de los puntos, necesarios para el agrupamiento, sin embargo, sólo utiliza los vecinos cercanos como lo sugiere el autor [45]. Lo cual difiere respecto a *MeanShift*, que codifica los datos de posición y movimiento de los puntos, como una medida que representa cada uno de los puntos. Además, el método de grafos no garantiza que se asigne a cada punto un grupo, es decir, puede ignorar algunos puntos de interés en su agrupamiento.

Conclusión La implementación realizada del algoritmo de seguimiento adaptativo de Stalder en [45], a diferencia de los autores, utiliza un muestreo denso de puntos para optimizar el costo computacional y evalúa tres diferentes métodos de agrupamiento. De acuerdo a los análisis realizados, se puede definir el método de *Mean Shift* como el más adecuado para la tarea de agrupamiento en este tipo de condiciones. De acuerdo a este estudio y el análisis realizado, se puede concluir que la medida

asociada a los puntos de objeto, *Target Score*, es la que mejor captura información de movimiento y apariencia (con el modelo SIFT), generando mejores resultados.

Capítulo 3

Modelo de Seguimiento Propuesto

El enfoque de este trabajo es seguir un elemento en una secuencia de video, independiente de la categoría de objeto. Y se pretende definir cada objeto por su contorno, en lugar de un cuadro o elipse de formas predefinidas. A partir de la revisión de los trabajos relacionados y las implementaciones de algunos de estos algoritmos, se ha planteado abordar esta investigación de acuerdo al diseño propuesto en el diagrama de la Figura 3.1. Esta técnica de seguimiento plantea una etapa de detección de puntos de interés y su propagación a través de la secuencia del video. Posteriormente, una etapa de agrupamiento de puntos basado en afinidad de distancia y movimiento. Por otro lado, se realiza una segmentación de la imagen para obtener pequeñas regiones de la misma que puedan conformar el objeto, estos segmentos se denominan superpíxeles. Luego, a partir de los puntos y de otros descriptores de apariencia y movimiento,se calculan unas medidas que indican el potencial de cada superpíxel de ser objeto o fondo. A partir de estos potenciales, se construye una función de energía que representa un grafo formado por superpíxeles como nodos. Finalmente, se realiza una optimización de la energía del grafo para obtener el etiquetado de cada superpíxel, entre Objeto y Fondo, con el cual se define el objeto de interés.

Describiendo de manera general el algoritmo, se inicializa manualmente la posición del objeto y se detectan puntos de interés para el primer frame, luego se implementa un ciclo que recorre el video frame a frame, en donde cada iteración se ejecuta el procedimiento esquematizado en el diagrama anterior. De esta forma, se

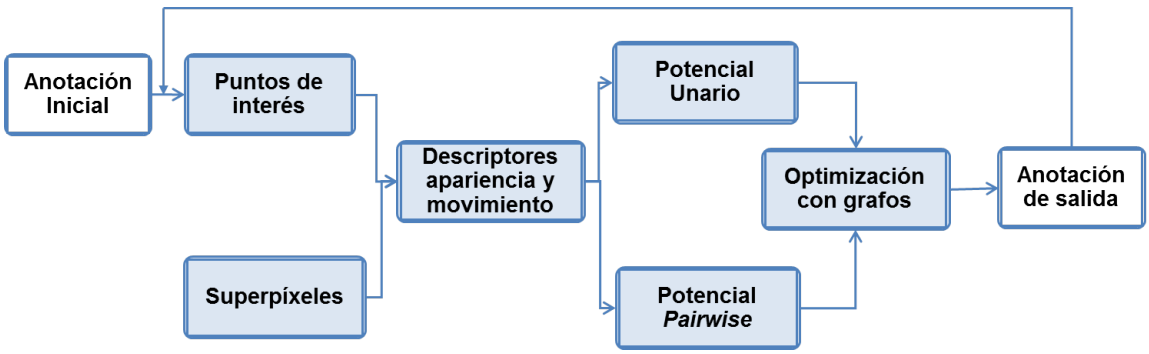

Figura 3.1: Esquema general del modelo de seguimiento de objetos propuesto

tiene al final de cada iteración una localización del objeto de interés. En la Tabla 3 se muestra el algoritmo implementado de manera resumida.

Tabla 3.1: Algoritmo de Seguimiento

Algoritmo de Seguimiento	
1	Localización Manual del Objeto
2	Inicio Ciclo
3	Detector de puntos de interés
4	Cálculo de descriptores SIFT para los puntos
5	Construcción del modelo de objetos basado en SIFT
6	Estimación de movimiento
7	Propagación de puntos al siguiente frame
8	Agrupamiento de puntos por afinidad de posición y movimiento
9	Segmentación de Imagen en superpíxeles
10	Cálculo de Potencial Unario (GMM, HOOF, TS)
11	Cálculo de Potencial Pairwise (Histograma 3D)
12	Construcción de la función de energía del grafo
13	Optimización basada en *Graph-Cut*
14	Etiquetado Objeto/fondo de salida
15	Fin Ciclo

3.1. Puntos de Interés

3.1.1. Detección y Propagación

El algoritmo de seguimiento se inicia con una localización manual del objeto a seguir por parte del usuario, esto para crear una máscara de contorno (binaria) en la cual se indica con 1 los píxeles que pertenecen al objeto y con 0 los que corresponden al fondo o escena. Seguidamente, se realiza una detección de puntos de interés, utilizando el detector de esquinas de Harris. Sólo se consideran algunos puntos de interés, en lugar de tomar cada uno de los pixeles, con el fin de optimizar el algoritmo en términos de costo computacional. Teniendo en cuenta que al utilizar un muestreo de puntos no se pierde información de toda la escena, sino que aplicar un detector de esquinas permite que se centre la atención en aquellos puntos de la imagen que puedan proveer información más relevante para el modelo propuesto. Es necesario, utilizar puntos en toda la escena y no enfocarse sólo en la región anotada como objeto; debido a que el sistema pretende construir un modelo tanto de objeto como de fondo. Esto para darle mayor robustez al modelo y mantener su desempeño en situaciones de oclusión, dado que en un caso de estos si no se analiza toda la imagen no se podría localizar el objeto cuando aparezca nuevamente.

Figura 3.2: Puntos de interés detectados en el primer frame de un video

Teniendo el conjunto de puntos detectados, se clasifican entre puntos de 'objeto' y puntos de 'escena', esto se obtiene discriminando entre unos y otros por medio de la máscara inicial. En la Figura 3.2 se observa un frame de una secuencia de video, indicando el contorno que contiene el objeto, así como los puntos de interés detectados, señalizados en azul como escena y en rojo como objeto.

A continuación, se utiliza un ciclo para recorrer frame a frame el video, en cada iteración lo primero que se realiza es la estimación de movimiento entre el frame anterior y el frame actual ($frame = i - 1$, $frame = i$). Esto se realiza por medio del cálculo del flujo óptico (*Optical flow*), el cual describe el vector de desplazamiento más probable que puede presentar cada punto de un frame en el siguiente instante de tiempo. Luego, se propagan los puntos detectados en el frame anterior hasta el frame actual, desplazándolos de acuerdo al valor obtenido en la estimación previa. La Figura 3.3 muestra el resultado de la predicción de movimiento para un par de frames; en rojo se muestran los puntos en el frame anterior y en azul la posición en la que quedarían esos puntos en el siguiente frame, las líneas en verde muestran los vectores de movimiento estimado.

Figura 3.3: Líneas de flujo óptico que representan el posible desplazamiento de los puntos de un frame al siguiente

Luego de propagar los puntos al siguiente frame, se aplica el detector para encontrar puntos 'Nuevos'. Para identificar si estos puntos pertenecen al conjunto de objeto o de escena, se compara con un modelo de objeto previamente creado. Este modelo es el conjunto de los descriptores SIFT (*Scale-Invariant feature transform*) calculados por cada punto de objeto. En la Figura 3.4 se muestran los puntos propagados hacia el siguiente frame de acuerdo a la estimación de movimiento. Estos puntos están diferenciados como objeto en rojo, escena en azul y nuevos en verde.

Figura 3.4: Puntos de interés detectados, clasificados como objeto, escena o nuevos, en el frame siguiente

3.1.2. Agrupamiento

Luego de tener el seguimiento de los puntos pertenecientes al objeto de interés, se realiza una actualización del objeto. Para ello se plantea un método de agrupamiento de puntos de tal forma que se obtengan grupos de puntos consistentes, que pueden ayudar a caracterizar regiones para identificarlas como un objeto. Estos puntos se agrupan de acuerdo a su distancia y movimiento, por medio de una

matrix de afinidad, utilizando un algoritmo basado en el método de *Mean Shift*, obteniéndose regiones consistentes que posiblemente puedan indicar la ubicación de un objeto. Esta técnica se realiza con el fin de actualizar el modelo del objeto de interés, adaptándolo a diferentes puntos de vista del objeto, oclusiones o cambios de intensidad. En la Figura 3.5 se muestra el resultado del agrupamiento de puntos, donde cada grupo formado se grafica con un color diferente.

Figura 3.5: Resultado del agrupamiento de puntos por *Mean Shift*

En esta etapa se computa una medida, que fortalece el modelo de seguimiento desarrollado. Esta medida corresponde al descriptor basado en puntos utilizado en el potencial unario del modelo de energía. En la Sección 3.3.1 se explica con detalle este potencial, el cual se utiliza para describir la función de energía del grafo creado a partir de superpíxeles. El valor obtenido permite capturar la relación entre los puntos etiquetados como objeto y aquellos etiquetados como fondo, por cada grupo. Donde obtener un mayor valor implica la presencia de puntos de objeto dentro de un mismo grupo.

3.2. Superpíxeles

El siguiente paso del algoritmo consiste en segmentar la imagen (frame actual) en regiones pequeñas, que denominamos superpíxeles, a partir de las cuales se puede obtener información importante para describir el objeto y seguirlo adecuadamente. La segmentación se implementa de acuerdo a una técnica utilizada comúnmente en visión por computador por sus óptimos resultados, denominada SLIC. En la Figura 3.6, se muestra el resultado de la segmentación, donde cada superpíxel obtenido, representa la unidad básica de análisis en el modelo propuesto.

Figura 3.6: Segmentación de la imagen. Los segmentos o superpíxeles, se representan por el promedio de color de los píxeles que contiene cada uno

3.3. Optimización con Grafos

Este trabajo aborda la tarea de seguimiento visual por medio de una técnica basada en grafos. Esta consiste en asignar un nodo a cada unidad de la secuencia de imágenes, en este caso los superpíxeles, y describir una función de energía que

caracterice el grafo formado por cada segmento y los enlaces que los unen entre sí. Y por medio de un método de optimización, se obtienen las etiquetas de objeto o fondo, para cada nodo, que minimizan la energía. El modelo del grafo consiste en un nodo por cada superpíxel y un enlace entre cada nodo y sus vecinos. La Figura 3.7 muestra la imagen segmentada de un frame, donde cada punto corresponde a un nodo o superpíxel y las líneas son los enlaces entre vecinos.

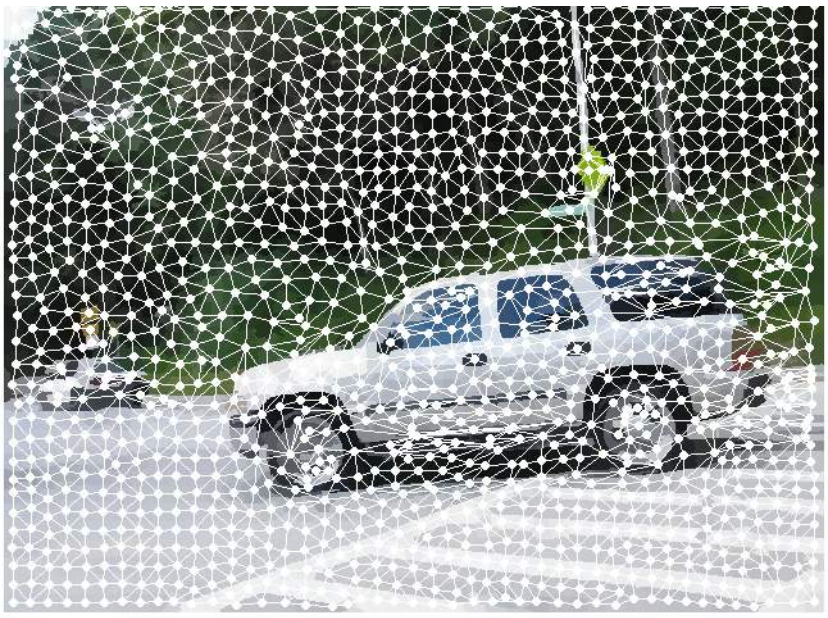

Figura 3.7: Imagen segmentada visualizando los nodos y enlaces del grafo construido

La construcción del modelo gráfico parte de los conceptos matemáticos que se representan en la siguiente expresión. La cual representa la energía del grafo, que se quiere minimizar.

$$E = \sum_{s_i \in S} \Psi(s_i|c_i) + \omega \sum_{s_i \in S} \Phi(s_i, s_j|c_i, c_j) \tag{3.1}$$

Donde $\Psi(s_i|c_i)$ corresponde al término unario, el cual es el potencial de un superpíxel para pertenecer a una clase (fondo, objeto). Y la función $\Phi(s_i, s_j|c_i, c_j)$ es el potencial *pairwise* o de pares que representa la conectividad de los nodos del grafo. A continuación se describe cada uno de los potenciales implementados.

3.3.1. Potencial Unario

El potencial unario es una medida que define las características de cada nodo, en este caso superpíxel, para pertenecer a una determinada clase, ya sea objeto o fondo. Este valor se determina a partir de características de apariencia y movimiento de cada segmento, en conjunto con una medida de pertenencia de puntos de objetos a cada región. Este potencial se computa de acuerdo a la siguiente expresión, que combina las diferentes características consideradas.

$$\Psi(s_i|c_i) = \alpha\Upsilon^s(s_i|c_i) + (1-\alpha)U^p(s_i|c_i) \tag{3.2}$$

Donde α es un parámetro de compensación establecido por medio de experimentos, para el equilibrio entre las diferentes características. La variable U^p relaciona la pertenencia de puntos de interés de objeto en cada superpíxel. Mientras que Υ^s resulta ser el término relacionado a la apariencia del superpíxel, que se define como la suma ponderada de dos términos, el costo de color y movimiento.

$$\Upsilon^s(s_i|c_i) = \beta U^c(s_i|c_i) + (1-\beta)U^m(s_i|c_i) \tag{3.3}$$

Siendo β el parámetro de compensación entre el costo de color U^c y el costo de movimiento U^m. Estos costos se determinan como la distancia promedio entre el descriptor del nuevo segmento y los descriptores de segmentos pertenecientes al modelo de objeto y al de fondo. En la Figura 3.8 se observa una representación del término que corresponde a las distancias de cada superpíxel al modelo aprendido para fondo (izquierda) y objeto (derecha).

Figura 3.8: Representación para un frame de las distancias de cada potencial al modelo de fondo (izquierda) y al modelo de objeto (derecha)

De acuerdo a la barra de color, los nodos con tonos hacia los azules, presentan un menor valor, es decir menor distancia de su descriptor al respectivo modelo. Según esto, se observa en la imagen de la derecha una silueta de un carro en azul, lo que indica que los superpíxeles de esa región son más similares al modelo de color del objeto.

Descriptor de Apariencia

Para describir cada superpíxel a partir de su característica de color, se construye un modelo de color para la clase fondo y uno para la clase objeto. Este par de modelos se aprende a partir de los segmentos etiquetados como objeto, o fondo, en el frame anterior; además, a través de la secuencia se va combinando los nuevos modelos computados. El modelo mixto gaussiano GMM (*Gaussian Mixture Model*) es el método implementado para esta tarea. Este consiste en una combinación de varias gaussianas que en conjunto permiten ajustarse más a la distribución de los datos. En la figura 3.9 se muestra una distribución de gaussianas en tres dimensiones, que corresponde a cada campo del espacio de color RGB. La combinación ponderada de tres componentes de gaussianas, se representa en la figura por los tres picos generados en la superficie.

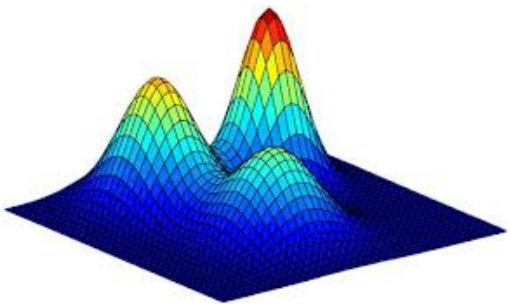

Figura 3.9: Representación del Modelo Mixto Gaussiano - GMM

El potencial obtenido corresponde a la probabilidad a posteriori de cada superpíxel de pertenecer al modelo GMM de objeto y de fondo, en cada caso. La Figura 3.10, presenta el valor del término relativo a apariencia para cada nodo, utilizando el modelo de color con GMM. En la imagen de la derecha, se muestra que los superpíxeles pertencientes al carro, se notan más cercano a los tonos azules, para

el modelo de objeto; y más cercano a los tonos rojos para el modelo de fondo, en la imagen de la izquierda. Esto indica que el descriptor de apariencia modela correctamente el objeto de interés (y la escena para el modelo GMM de fondo), es útil para describir cada superpíxel en términos de apariencia y permite discriminar correctamente entre una clase y otra.

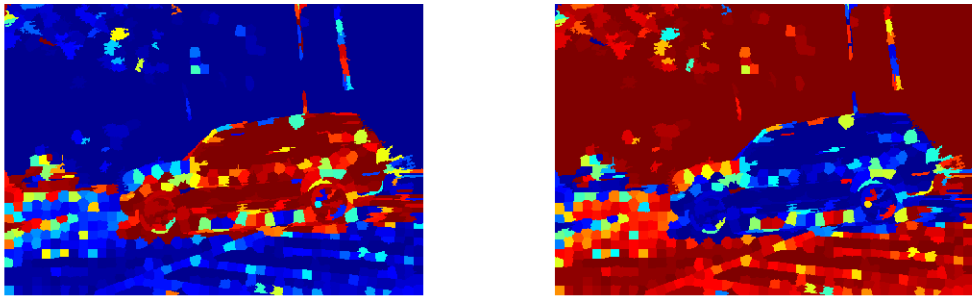

Figura 3.10: Representación para un frame de las distancias de cada valor relativo a la apariencia, al modelo de fondo (izquierda) y al modelo de objeto (derecha)

Descriptor de Movimiento

El flujo óptico es una medida que indica el movimiento más probable de cada píxel de un frame al siguiente. Con esta medida se construye el Histograma de Flujo Óptico Orientado (HOOF - por sus siglas en inglés), de nueve *bins* como se muestra en la figura 3.11. Este histograma resulta ser el descriptor de movimiento que se implementa en este sistema, para construir el modelo de objeto y fondo.

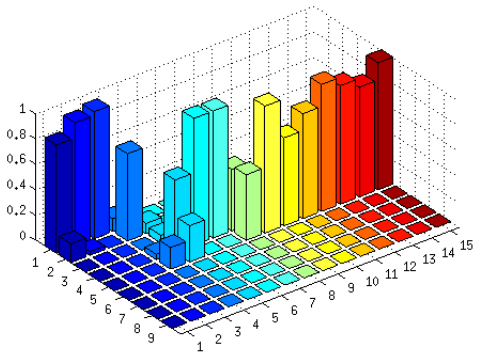

Figura 3.11: Histograma de Flujo Óptico Orientado - HOOF

La Figura 3.12 representa el término relativo a movimiento, para la construcción del potencial unario. Para este *frame* de la secuencia de Cars-MoSeg, se observa que esta medida no discrimina entre el objeto y el fondo. Sin embargo, en otros casos si es relevante este término. Además, según las investigaciones relacionadas el movimiento es una de las principales características para aportar información en la tarea de seguimiento.

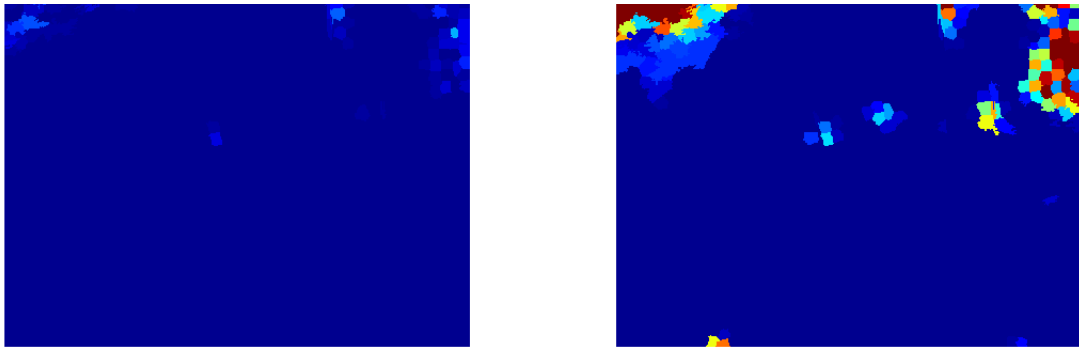

Figura 3.12: Representación para un frame de las distancias de cada valor relativo a movimiento, al modelo de fondo (izquierda) y al modelo de objeto (derecha)

Descriptor basado en Puntos

A partir de los superpíxeles y los grupos de puntos encontrados se calcula una métrica tomada de [45], la cual permite cuantificar qué tan posible una región puede pertenecer a un objeto, independiente de la categoría. Esta métrica TS favorece aquellos superpíxeles que contengan más puntos de objeto que de escena o fondo.

$$TS(s) = \frac{Num_{obj}(s) - Num_{esc}(s)}{Num(s)} \tag{3.4}$$

Donde Num_{obj} es el número de puntos de objetos en cada segmento, Num_{esc} es el número de puntos de escena en cada superpíxel. Esta medida se normaliza, para que cada término del potencial unario corresponda a un valor entre 0 y 1, donde 0 indica una alta pertenencia del segmento a una clase. El valor obtenido con esta métrica representa un porcentaje de la cantidad de puntos de objetos en un segmento, por tanto se utiliza en la siguiente expresión para conformar el término unario relativo a puntos de interés U^p.

$$U^p(s_i|c_i) = \begin{cases} TS, & c_i = 0 \\ 1 - TS, & c_i = 1 \end{cases} \tag{3.5}$$

En la siguiente imagen, se muestra una representación del valor descrito anteriormente, relativo a puntos de interés, respecto al fondo y al color.

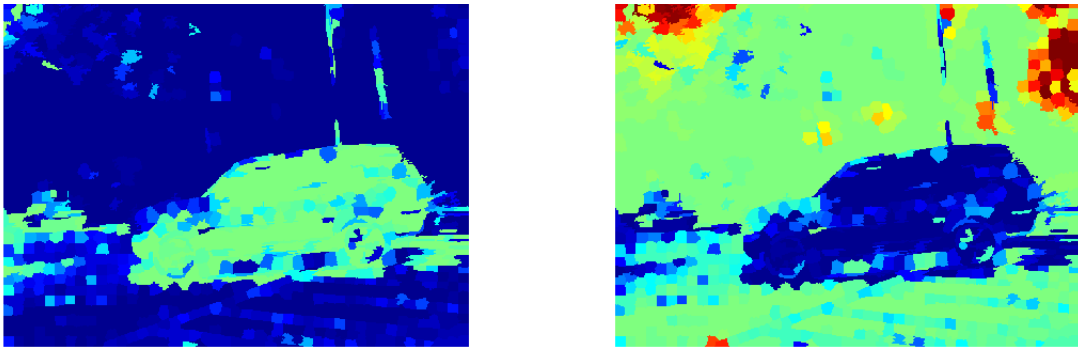

Figura 3.13: Representación para un frame de las distancias de cada potencial relativo a los puntos de interés, al modelo de fondo (izquierda) y al modelo de objeto (derecha)

3.3.2. Potencial *Pairwise*

En la construcción del grafo es necesario definir el valor de los enlaces entre nodos, de esta forma, es posible establecer prioridades entre los nodos de tal forma que se puedan clasificar correctamente entre las dos clases, objeto o fondo. Este valor es el Potencial *Pairwise* o de parejas, y corresponde a la similitud entre el descriptor de apariencia de cada superpíxel y el descriptor de cada superpíxel vecino. Este potencial se define según la expresión en 3.6. En la cual se muestra que la distancia computada entre los descriptores de cada superpíxel, es transformada en una medida de similitud. Además, se define este potencial como un valor de relación entre vecinos cercanos, esto se refiere a que los únicos enlaces presentes en el grafo y utilizados para la optimización, son aquellos que unen un nodo o superpíxel con el vecino espacial.

$$\Phi(s_i, s_j|c_i, c_j) = e^{-d(\Upsilon_i, \Upsilon_j)}[c_i \neq c_j], \qquad \forall s_i \sim^n s_j \tag{3.6}$$

Donde, $s_i \sim^n s_j$ indica dos segmentos vecinos, como se simboliza en [40].

Descriptor de color

Para describir cada segmento, se utiliza un histograma de color de 3 dimensiones, donde cada dimensión representa un campo del espacio de color RGB. En este histograma, cada *bin* representa un rango de colores del espacio RGB y la frecuencia corresponde al número de píxeles que tienen un valor de color dentro de ese rango. Para cada superpíxel se computa un histograma 3D, que se asocia al modelo de color para Objeto o Fondo, según la anotación inicial o la salida del algoritmo en el *frame* anterior.

En la figura 3.14, se observa el histograma utilizado, la imagen de la izquierda muestra el histograma para todos los posibles elementos del descriptor, donde cada esfera es un *bin* y su color representa el valor medio de los colores que caen en el rango de dicho *bin*. En la imagen de la derecha, se observa el histograma resultante para un superpíxel, en el cual sólo existen algunas esferas activadas, indicando con su tamaño la frecuencia de ocurrencia de cada píxel en el rango de color determinado por cada esfera.

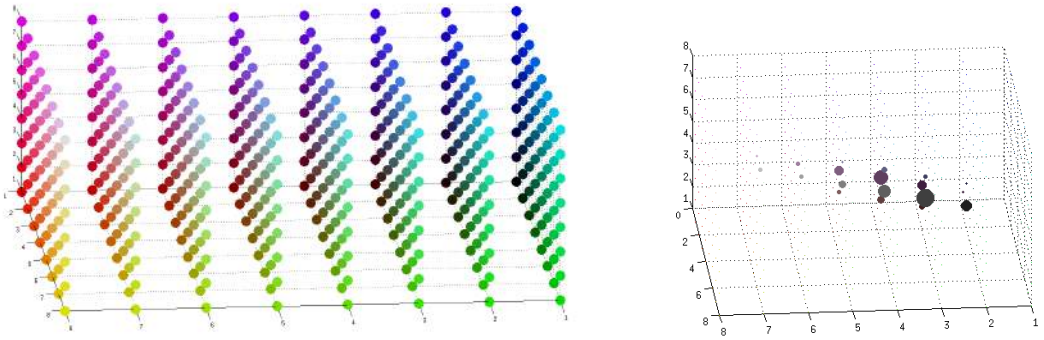

Figura 3.14: Descriptor de color. Histograma en 3D representando la gama de colores que puede tomar un píxel (Izquierda). Histograma RGB en 3D para un superpíxel(Derecha)

Capítulo 4

Experimentos y Resultados

En este capítulo se demuestra la importancia del modelo de seguimiento propuesto para obtener un buen desempeño en la tarea de seguimiento visual. Esto se realiza por medio de un conjunto de experimentos y análisis cuantitativos y cualitativos.

Inicialmente, se describen los videos utilizados para evaluación del sistema, que corresponden a tres conjuntos de videos de distintos contextos. Luego, se especifican los detalles de implementación de cada etapa del algoritmo. En la siguiente sección se expone el experimento para selección de parámetros y los resultados obtenidos. La cuarta sección corresponde a los resultados cuantitativos, es aquí donde se definen los experimentos de verificación del sistema, para demostrar la relevancia del modelo propuesto; además, se presentan las comparaciones cuantitativas con investigaciones relacionadas para validación del desempeño del sistema. Finalmente, se muestran los resultados cualitativos, donde se presentan imágenes del seguimiento visual realizado por el algoritmo en algunas secuencias utilizadas.

4.1. Conjunto de Videos

El seguimiento de objetos, como toda tarea de visión por computador requiere una etapa de adquisición de datos. Las cámaras y sensores de movimiento son las herramientas utilizadas para capturar los datos que serán objeto de estudio y a partir de los cuales se construyen una colección o base de datos (dataset). Cada colección de videos o imágenes recolectados se utiliza para diferentes tareas de acuerdo a

las características que tenga. Esto es, seguimiento de objetos, si hay un elemento claramente en movimiento, reconocimiento de escenas, si las imágenes corresponden a paisajes o ambientes específicos, entre otras. La libre distribución de los datos entre los investigadores, permite evaluar y realizar comparaciones justas y equitativas para fortalecer el trabajo en este tema de estudio.

Analizando las aplicaciones de un modelo de seguimiento visual, tales como sistemas de vigilancia, videojuegos, monitoreo de tráfico vehicular, entre otros, se ha definido los contextos que se desean abordar en esta investigación. Se recolectaron tres conjuntos de videos de distintos contextos, con su correspondiente *ground truth*, el cual consiste en una máscara definiendo el contorno del objeto de interés, para determinados frames de la secuencia.

Cars-MoSeg El primer conjunto de videos se asocia a las aplicaciones de monitoreo de tráfico vehicular, por lo cual las secuencias son tomadas de vehículos en carreteras, con diferentes puntos de vista, con diferentes condiciones de iluminación, con movimiento de cámara, entre otras condiciones variables.

Este conjunto de videos lo forman 10 secuencias cortas, con su correspondiente anotación manual o *ground truth*. Estos videos hacen parte del "Berkeley Motion Segmentation Dataset" (MoSeg), disponible en internet [48], en la Figura 4.1 se muestran algunas de estas secuencias.

Figura 4.1: Videos del contexto de tráfico vehicular

SegTrack El segundo conjunto consiste en videos en los que se tienen personas o animales en ambientes reales, en escenas simples; este se tomó de la base de datos SegTrack, presentada en [49] y disponible en internet en [50]. Este conjunto de videos

está formado por seis secuencias de corta duración, con su correspondiente *ground truth*.

Sin embargo, por recomendación de los trabajos relacionados, sólo se utilizan cinco secuencias de videos, excluyendo la denominada *Penguin*, debido a la ambiguedad que puede generar identificar el objeto a seguir, al tener un grupo de pinguinos en la escena. La Figura 4.2 muestra imágenes del segundo conjunto recolectado.

Figura 4.2: Videos de personas y animales en escenas simples

MCCD El tercer contexto corresponde a obras de construcción civil, en las cuales la tarea de seguimiento permitiría monitoreo continuo del progreso de la construcción, localización de maquinaria, de la cual se obtienen videos de maquinarias pesadas, como volquetas, excavadoras, camiones, carros, en fondos de construcción en campo abierto.

La Figura 4.3, muestra imágenes del conjunto que se ha denominado *MCCD* (*Machines in Civil Construction Dataset*), recolectado en colaboración de investigadores de la Universidad de Illinois en Urbana-Champaign. En total, se tienen 10 secuencias de video cortas.

Figura 4.3: Videos relativos a maquinarias en obras de construcción civil

4.2. Métricas de desempeño

La tarea de seguimiento, se puede evaluar cuantitativamente comparando la posición original definida por el usuario en ciertos frames, con la posición entregada por el algoritmo implementado, la cual se define por medio de una máscara de contorno. Esta comparación se realiza con la medida de solapamiento (*overlap*) entre ambas salidas, que mide la razón entre la intersección y la unión de estas regiones.

$$\text{solapamiento} = \frac{\text{segmentoFG} \cap groundtruth}{\text{segmentoFG} \cup groundtruth} \tag{4.1}$$

Donde segmentoFG corresponde a la región obtenida por el algoritmo como objeto y *groundtruth* es la región manualmente anotada por el usuario. Esta medida permite obtener una precisión a nivel de píxel, lo cual permite evaluar el desempeño de manera estricta.

El número de píxeles etiquetados erróneamente, es otra medida que evalúa el desempeño a nivel de píxel, utilizada por los investigadores del tema, como Lee en [34]. Para definir esta métrica, es necesario indicar que corresponde a Verdadero Positivo, Verdadero Negativo, Falso Positivo y Falso Negativo, en el contexto de seguimiento articulado.

Verdadero Positivo: Número de píxeles de objeto correctamente etiquetados como "Objeto".
Verdadero Negativo: Número de píxeles de fondo correctamente etiquetados como "Fondo".
Falso Positivo: Número de píxeles de objeto etiquetados erróneamente como "Objeto".
Falso Negativo: Número de píxeles de fondo etiquetados erróneamente como "Fondo".

De acuerdo a estas definiciones, la medida que tomaremos para evaluar el desempeño del sistema, resulta ser la combinación de los píxeles que el algoritmo indicó de manera equivocada que eran objeto, con los píxeles que asignó erróneamente como

fondo. Lo cual se representa en la siguiente expresión.

$$\text{Píxeles Erróneos} = \text{Falso Positivo} + \text{Falso Negativo} \tag{4.2}$$

Para tener una medida de píxeles erróneos, que permita analizar y compararse entre cualquier secuencia de video, se utiliza el porcentaje de píxeles erróneos, como una normalización del número de píxeles erróneos, determinado como se muestra a continuación.

$$\%\ \text{Píxeles Erróneos} = \frac{\text{Píxeles Erróneos}}{\text{Total de Píxeles}} \tag{4.3}$$

4.3. Detalles de implementación

El código está desarrollado de acuerdo al algoritmo de la Tabla 3.1, cumpliendo con los requerimientos establecidos en la etapa de diseño. A continuación se definen cada una de las técnicas y parámetros utilizados en el algoritmo de seguimiento.

La implementación del algoritmo se ha realizado en la herramienta Matlab, en conjunto con algunas librerías obtenidas en C/C++. La detección de puntos de interés se realiza con un método basado en el mínimo valor propio o *Eigenvalue*, que captura máximo 1000 puntos por frame. La estimación por movimiento se obtiene por medio del flujo óptico, utilizando una función de la librería de Piotr, disponible en [51]. En este algoritmo se utiliza el método de "LK" basado en el trabajo de [21]. Para tener una estimación más robusta, se computa el flujo óptico hacia adelante y hacia atras y se acepta el resultado que ofrezca mayor fiabilidad . El descriptor SIFT se computa con la librería VLFeat, disponible en [52], frame a frame centrado en la posición de cada punto, con una escala de 10.

4.3.1. Selección de Parámetros

En el modelo propuesto se plantea una función de energía que representa el grafo constituido para etiquetar cada superpíxel como objeto o fondo. Esta función esta construida como la suma ponderada de diferentes términos, unarios y *pairwise*, definidas en las ecuaciones 3.1, 3.2, 3.3; donde se combinan características de apariencia,

movimiento y basadas en puntos. Los parámetros α, β y ω son utilizados para modificar la importancia de cada característica en la función de energía. Para seleccionar de manera adecuada de estos valores, se realiza un diseño de experimentos, que se describe a continuación.

Hipótesis : El desempeño del sistema de seguimiento de objetos se ve influenciado por la combinación de los parámetros α, β y ω.
Variable de Respuesta: Solapamiento de la salida del algoritmo con el *ground truth*.
Factores: α, β y ω.
Niveles:

$\alpha = [0{,}25\ \ 0{,}5\ \ 0{,}75\ \ 1]$
$\beta = [0{,}25\ \ 0{,}5\ \ 0{,}75\ \ 1]$
$\omega = [0{,}5\ \ 1\ \ 1{,}5\ \ 2]$

Número de observaciones: Cantidad de videos utilizados, diez (10).
Experimento: Para analizar este planteamiento, se utiliza un diseño factorial multinivel. Teniendo un total de experimentos de $NTE = 4 * 4 * 4 * 10 = 640$.

En la ejecución del diseño de experimentos, se utilizan los 10 videos del conjunto Cars-BMS26. A partir de los resultados obtenidos, se busca el máximo solapamiento, para establecer los parámetros utilizados en el modelo propuesto. Los cuales resultan ser los valores mostrados en la Tabla 4.1.

Tabla 4.1: Parámetros seleccionados por experimentación en cada video

Video	**Cars1**	**Cars2**	**Cars3**	**Cars4**	**Cars5**
α	0.5	0.25	0.25	0.25	0.25
β	0.5	0.5	1	0.25	1
ω	1	0.5	1	0.5	0.5

Video	**Cars6**	**Cars7**	**Cars8**	**Cars9**	**Cars10**
α	0.25	0.75	0.5	0.25	0.5
β	0.75	0.75	0.5	0.5	0.25
ω	0.5	0.5	0.5	0.5	0.5

4.4. Verificación del Desempeño

4.4.1. Desempeño del modelo de color

El sistema propuesto establece un descriptor de apariencia que representa el color de los superpíxeles, por medio de un Modelo Mixto Gaussiano - GMM, aprendido en cada frame. Para demostrar la relevancia del GMM en el sistema propuesto, se compara con una implementación equivalente que utiliza un modelo de histogramas 3D. Este modelo es el conjunto de los histogramas de color 3D, de los superpíxeles etiquetados en el frame anterior. Para computar el potencial unario, se obtiene la mínima distancia entre el histograma del nuevo superpíxel y cada histograma del modelo construido.

Tabla 4.2: Desempeño en el conjunto de videos Cars-MoSeg

Video	**Cars1**	**Cars2**	**Cars3**	**Cars4**	**Cars5**
Histograma 3D	26.31 %	18.25 %	26.49 %	44.54 %	31.82 %
GMM	78.67 %	49.38 %	54.68 %	18.71 %	32.80 %

Video	**Cars6**	**Cars7**	**Cars8**	**Cars9**	**Cars10**
Histograma 3D	0 %	0 %	0 %	11.45 %	0 %
GMM	5.43 %	3.58 %	45.79 %	44.28 %	32.37 %

La Tabla 4.2 muestra el desempeño del sistema en el conjunto de videos Cars-MoSeg utilizando cada uno de los dos modelos, ambos con los mismos parámetros definidos anteriormente. Para el modelo que utiliza histogramas 3D se tiene un promedio general de 15,89 % y con el modelo de color basado en GMM se obtuvo 36,57 %. Se demuestra cuantitativamente la importancia del modelo construido por medio de un conjunto de gaussianas aprendidas a través de toda la secuencia de imágenes, frente a un modelo aprendido únicamente en el frame anterior. Por medio del modelo GMM se almacena información del objeto desde los primeros frames, con lo cual se tiene un sistema robusto ante posibles oclusiones.

4.4.2. Desempeño del Sistema

El sistema de seguimiento visual de objetos se evaluó en tres conjuntos de videos, de diferentes contextos, los cuales se describieron anteriormente. Los parámetros para los otros dos conjuntos de videos se seleccionan como se hizo para el conjunto de Cars-MoSeg. A continuación se muestra el desempeño del sistema en función del porcentaje de solapamiento y del número de píxeles etiquetados erróneamente, para cada uno de los contextos de los videos.

Tabla 4.3: Desempeño del sistema global en cada conjunto de videos

Conjunto de Video	**Cars-MoSeg**	**SegTrack**	**MCCD**
% solapamiento	36.57 %	17.48 %	53.72 %
# píxeles erróneos	54786	9181	7957
% píxeles erróneos	5.89 %	11.49 %	6.10 %

A continuación, se detallan los resultados obtenidos en cada uno de los contextos de videos utilizados, presentando el desempeño del sistema de seguimiento por video.

Cars-MoSeg

El conjunto de videos Cars-MoSeg, contiene 10 secuencias cortas con imágenes de vehículos en carretera, con movimiento de cámara y cambios de iluminación. En la siguiente tabla se presenta el desempeño para cada video. En estos datos se observa que el desempeño del sistema en general es bueno. Sin embargo, se tienen algunos casos porcentajes bajos, los cuales se asocian tanto a posibles errores en el etiquetado manual, a nivel de píxel, así como a la poca precisión por no tener anotaciones (*groundtruth*) en cada frame, dado que así está diseñado el conjunto de videos disponible en Internet.

Tabla 4.4: Desempeño en el conjunto de videos Cars-MoSeg

Video	**Cars1**	**Cars2**	**Cars3**	**Cars4**	**Cars5**
% solapamiento	78.67 %	49.38 %	54.68 %	18.71 %	32.80 %
# píxeles erróneos	10800	10520	9350	14630	16810
% píxeles erróneos	3.52 %	3.42 %	3.04 %	4.76 %	5.47 %

Video	**Cars6**	**Cars7**	**Cars8**	**Cars9**	**Cars10**
% solapamiento	5.43 %	3.58 %	45.79 %	44.28 %	32.37 %
# píxeles erróneos	13950	17547	12540	13810	60850
% píxeles erróneos	4.54 %	5.71 %	4.08 %	4.50 %	19.81 %

La Figura 4.4 contiene algunos *frames* del contexto de tráfico vehicular. Es necesario resaltar lo desafiante de este conjunto de videos, en el cual se tienen objetos de apariencia similar a varios elementos considerados como escena, como otros carros alrededor. Además, son videos con cambios de iluminación, cambios de punto de vista y movimiento de cámara, lo cual indica que la salida de este algoritmo es aceptable considerando el nivel de dificultad de los datos.

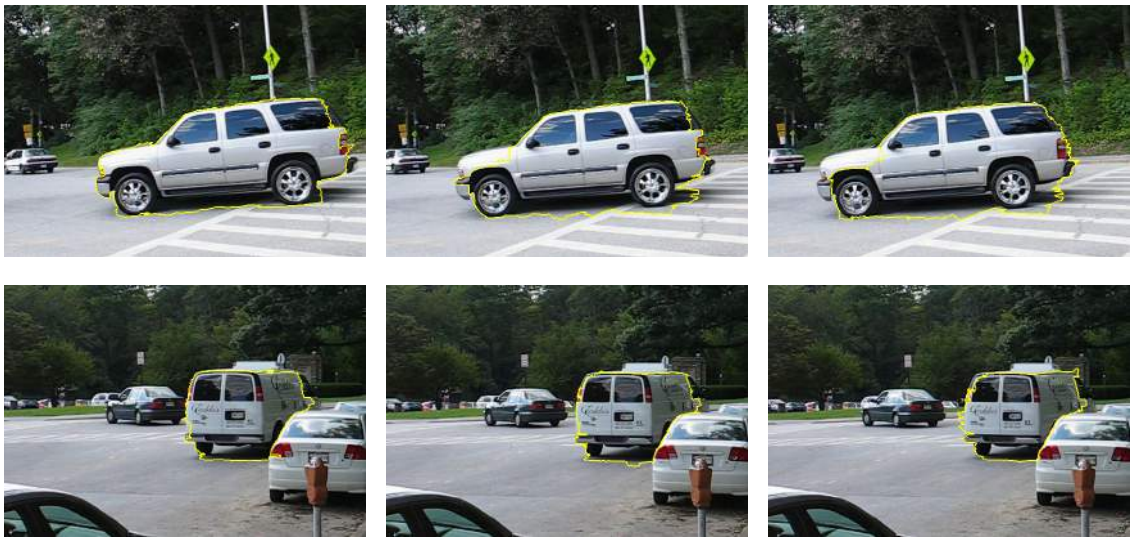

Figura 4.4: Resultados visuales del sistema de seguimiento por video para el conjunto Cars-Moseg

En las imágenes mostradas el contorno en amarillo indica la salida del algoritmo, la cual adopta la forma del objeto de interés de manera aproximada para la mayoría de los casos. En algunos ejemplos, se incluyen pequeños errores al seleccionar como objeto a regiones del fondo de la imagen; sin embargo, en el contorno obtenido se puede identificar la forma del carro, a pesar de los pequeños errores.

SegTrack

Para evaluar en el conjunto de videos SegTrack, se utilizaron los parámetros α, β y ω seleccionados en un experimento análogo al realizado en los videos de CarsMoSeg. La Tabla 4.5 presenta los resultados del modelo propuesto, para cada video del conjunto SegTrack. En estos videos, se tienen muchas escenas desafiantes, con similar apariencia entre el fondo y el objeto, lo cual se nota por tener resultados que no son muy altos. Sin embargo, estos resultados no implican que no se tiene un seguimiento del objeto y esto se respalda en los resultdos visuales mostrados en la siguiente sección.

Tabla 4.5: Desempeño en el conjunto de videos SegTrack

Video	**Birdfall**	**Cheetah**	**Girl**	**MonkeyDog**	**Parachute**
% solapamiento	0 %	4.91 %	18.90 %	3.45 %	43.84 %
# píxeles erróneos	567	1681	8882	35454	2192
% píxeles erróneos	0.66 %	2.18 %	6.94 %	46.16 %	1.50 %

En la figura 4.5, se tienen los resultados del algoritmo para algunas de las secuencias del conjunto SegTrack. Estas secuencias están conformadas por personas o animales en escenas con poco contraste de color, con movimientos rápidos y objetos articulados que cambian de forma a través del tiempo; lo cual lo hace un conjunto desafiante para la tarea de seguimiento.

En la primera fila de las imágenes se presenta un buen desempeño del algoritmo, mostrando una salida que se adapta correctamente a la forma del objeto de interés. En la secuencia inferior se muestra un mono seleccionado como el objeto de interés, el cual cambia rápidamente de movimiento; en este caso, el algoritmo selecciona el objeto pero lo confunde con algunas regiones de la escena que presentan apariencia similar, en cuanto a color se refiere; estos casos de falla son aceptables, considerando lo complejo de la escena debido a la presencia de un perro que interactúa con el mono y que puede distraer al algoritmo del objeto principal.

Es necesario explicar el resultado obtenido en la secuencia de *birdfall2*, en la cual el algoritmo falla completamente. Si se observa la secuencia mostrada en la Figura 4.6 donde se tiene un ave cayendo junto a un árbol, se nota que las escenas son

Figura 4.5: Resultados visuales del sistema de seguimiento por video para el conjunto SegTrack

complejas y la apariencia es uniforme, además de tener un objeto que no contrasta con el fondo. En este caso es claro el nivel de dificultad, ya que incluso una persona no identifica y localiza con precisión el objeto de interés sin una mirada detallada.

Figura 4.6: Frames de la secuencia de video Birdfall. En estas imágenes es difícil identificar el objeto de interés por la uniformidad de la escena

El modelo propuesto no es robusto ante situaciones críticas, como esta secuencia. En la figura 4.7 se muestra el potencial unario, relacionado al modelo de fondo y al modelo de objeto, el cual se nota sesgado a seleccionar cada superpíxel de la imagen como fondo. Esta representación permite concluir que la apariencia descrita por el GMM, el movimiento representado en el histograma HOOF o la medida de puntos de objeto relacionada al modelo SIFT que componen el potencial unario, no resultan ser suficientes para distinguir un objeto en escenas muy complejas.

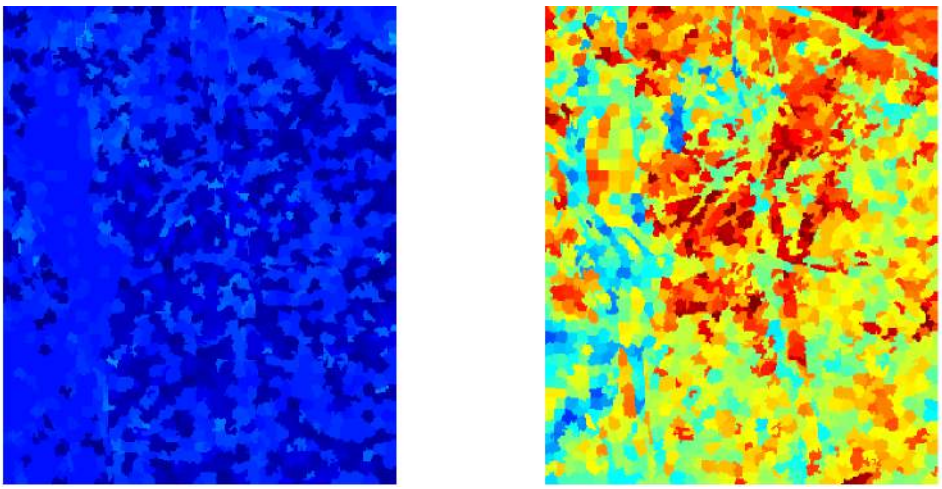

Figura 4.7: Potencial Unario de BirdFall2. Representación de las distancias de cada potencial al modelo de fondo (izquierda) y al modelo de objeto (derecha), para el video BirdFall2

MCCD

El tercer contexto considerado en los videos utilizados, es el de maquinarias de obras de construcción civil. En estas 10 secuencias cortas se evalúa el sistema de seguimiento, presentando el desempeño mostrado en la tabla 4.6. En estos resultados se muestran valores de solapamiento altos, como en las secuencias MCCD4 y MCCD5 o MCCD10. Esto indica que, a pesar de tener videos desafiantes, con escenas muy homogéneas en apariencia y color, con objetos articulados y cambios de iluminación, el algoritmo es capaz de seguir el objeto de interés y adaptarse correctamente a su contorno.

Tabla 4.6: Desempeño en el conjunto de videos MCCD

Video	**MCCD1**	**MCCD2**	**MCCD3**	**MCCD4**	**MCCD5**
% solapamiento	8.09 %	36.21 %	16.69 %	72.48 %	75.91 %
# píxeles erróneos	36061	9992	5524	2909	3207
% píxeles erróneos	27.62 %	7.65 %	4.23 %	2.23 %	2.46 %

Video	**MCCD6**	**MCCD7**	**MCCD8**	**MCCD9**	**MCCD10**
% solapamiento	65.84 %	66.95 %	57.60 %	50.80 %	86.65 %
# píxeles erróneos	3854	5229	3616	6074	3101
% píxeles erróneos	2.96 %	4.01 %	2.77 %	4.65 %	2.38 %

En las imágenes mostradas en la Figura 4.8, se tiene la salida del algoritmo para el conjunto de videos de de maquinarias en obras de construcción. Estas secuencias son realmente complicadas y resultan ser todo un reto para un algoritmo de seguimiento, considerando que se tienen escenas donde no existen fuertes contrastes de color, donde se tienen máquinas similares en el entorno y se tiene cambios de iluminación ya que son videos grabados al aire libre; además, estos objetos son articulados y cambian su forma cada vez que se mueven. Por lo cual se tiene que los resultados del sistema implementado son muy buenos, ya que muestran un seguimiento correcto que se adapta muy bien conforme al movimiento de los objetos.

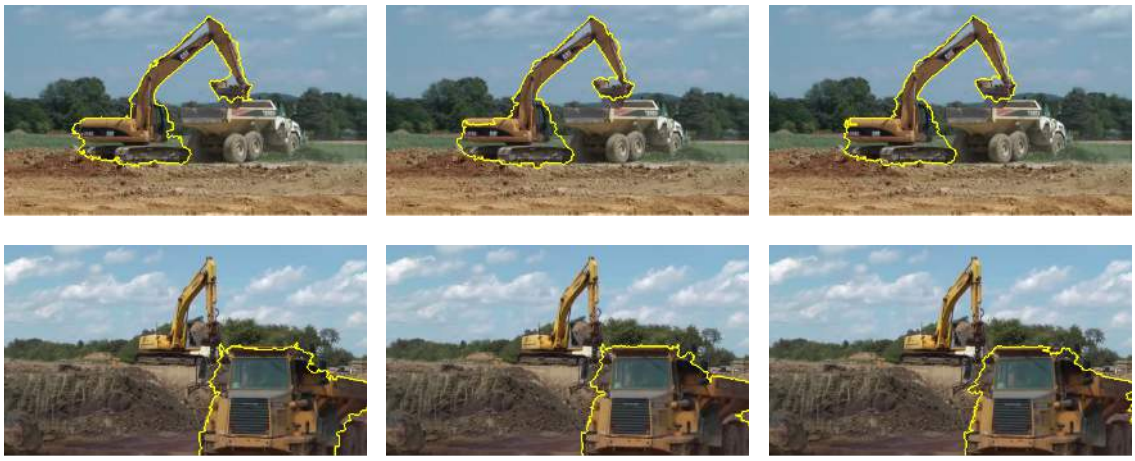

Figura 4.8: Resultados visuales del sistema de seguimiento para el conjunto MCCD

En la primera secuencia mostrada se tiene un buen desempeño, ya que el contorno se adapta con precisión al objeto de interés. En estos casos, el algoritmo es capaz de realizar correctamente la tarea de seguimiento a pesar de tener condiciones de apariencia muy uniformes en la escena. En la última secuencia se muestra un objeto de interés diferente a el de las otras secuencias; en este caso, también se obtiene un resultado muy preciso que se adapta a la forma del camión.

4.5. Validación del desempeño

Luego de verificar el funcionamiento del sistema de seguimiento visual, en diferentes conjuntos de datos, se compara el desempeño obtenido con trabajos relacionados. A continuación se presenta en la Taba 4.5 los valores obtenidos del Porcentaje de de Píxeles erróneos, por secuencia del conjunto SegTrack. En esta se puede notar que el sistema propuesto es comparable en desempeño con los actuales algoritmos de seguimiento de objetos.

Tabla 4.7: Comparación del desempeño del sistema con trabajos relacionados, en función del porcentaje de píxeles erróneos

# píxeles erróneos	Birdfall	Cheetah	Girl	MonkeyDog	Parachute
Chockalingam en [53]	0.54 %	1.58 %	1.37 %	0.89 %	0.34 %
Tsai en [49]	0.30 %	1.49 %	1.01 %	0.73 %	0.16 %
Lee en [34]	0.34 %	1.18 %	1.40 %	0.68 %	0.14
Tianyang en [[54]	0.22 %	1.05 %	0.13 %	0.62 %	0.15 %
Zhang en [55]	0.18 %	0.82 %	1.16 %	0.48 %	0.15 %
Brox en [56]	0.55 %	2.56 %	5.93 %	1.87 %	0.07 %
P. Ochs en [57]	0.55 %	1.53 %	4.44 %	1.87 %	1.09 %
Barnichm en [58]	0.72 %	14.60 %	20.63 %	16.49 %	27.62
Papazoglou en [59]	0.26 %	1.16 %	3.01 %	0.37 %	0.59 %
Sistema Propuesto	1.52 %	2.19 %	4.13 %	46.16 %	1.50

Esto indica que la propuesta de un modelo de grafos de superpíxeles, utilizando las características planteadas en esta investigación, entrega buenos resultados, ya que si se analiza cada secuencia, el modelo propuesto en esta investigación es mejor en

algunos casos o similares en otros. A pesar que se presentan algunos casos con menor rendimiento, existen otras características para tener en cuenta en la comparación. Es necesario resaltar que nuestro modelo a diferencia de los trabajos relacionados que se compararon, se puede desarrollar en aplicaciones en *streaming*, es decir, puede realizar el seguimiento en cada *frame* que vaya llegando al sistema; esto es, sin necesidad de procesar todo el video completo.

Para evaluar cualitativamente nuestro sistema se muestra en la figura 4.9, el resultado de algunos *frames* para una secuencia de video, para el modelo propuesto y el trabajo de [37]. En estas imágenes se puede observar que el desempeño del modelo propuesto es mejor que los resultados del trabajo de Sharir.

Figura 4.9: Comparación en Secuencia de imágenes de Cars-MoSeg. Fila superior, resultado del modelo propuesto. Fila inferior, resultado del trabajo de [37]

La figura 4.10 presenta una comparación visual del resultado de algunos *frames* para una secuencia de video, del conjunto SegTrack, entre el modelo propuesto y el trabajo de [59]. En la secuencia de *Parachute*, se obtienen resultados satisfactorios y precisos, obteniendo para ambos trabajos un contorno muy bien definido. En la segunda secuencia, *Girl*, el contorno definido por el modelo propuesto presenta menor precisión. De acuerdo a ambos resultados, se puede afirmar que los resultados son comparables. Pero es necesario resaltar que el modelo de seguimiento propuesto ofrece una ventaja, desde que su diseño permite realizar el procesamiento de la secuencia, *frame a frame*, a diferencia del trabajo de [59], el cual requiere procesar

todo el video completo para generar el seguimiento. Esta característica hace que el modelo propuesto sea más conveniente para múltiples aplicaciones, donde se requiere *streaming*. Como en vigilancia y monitoreo, donde los datos de video se reciben continuamente y se debe obtener un resultado inmediatamente.

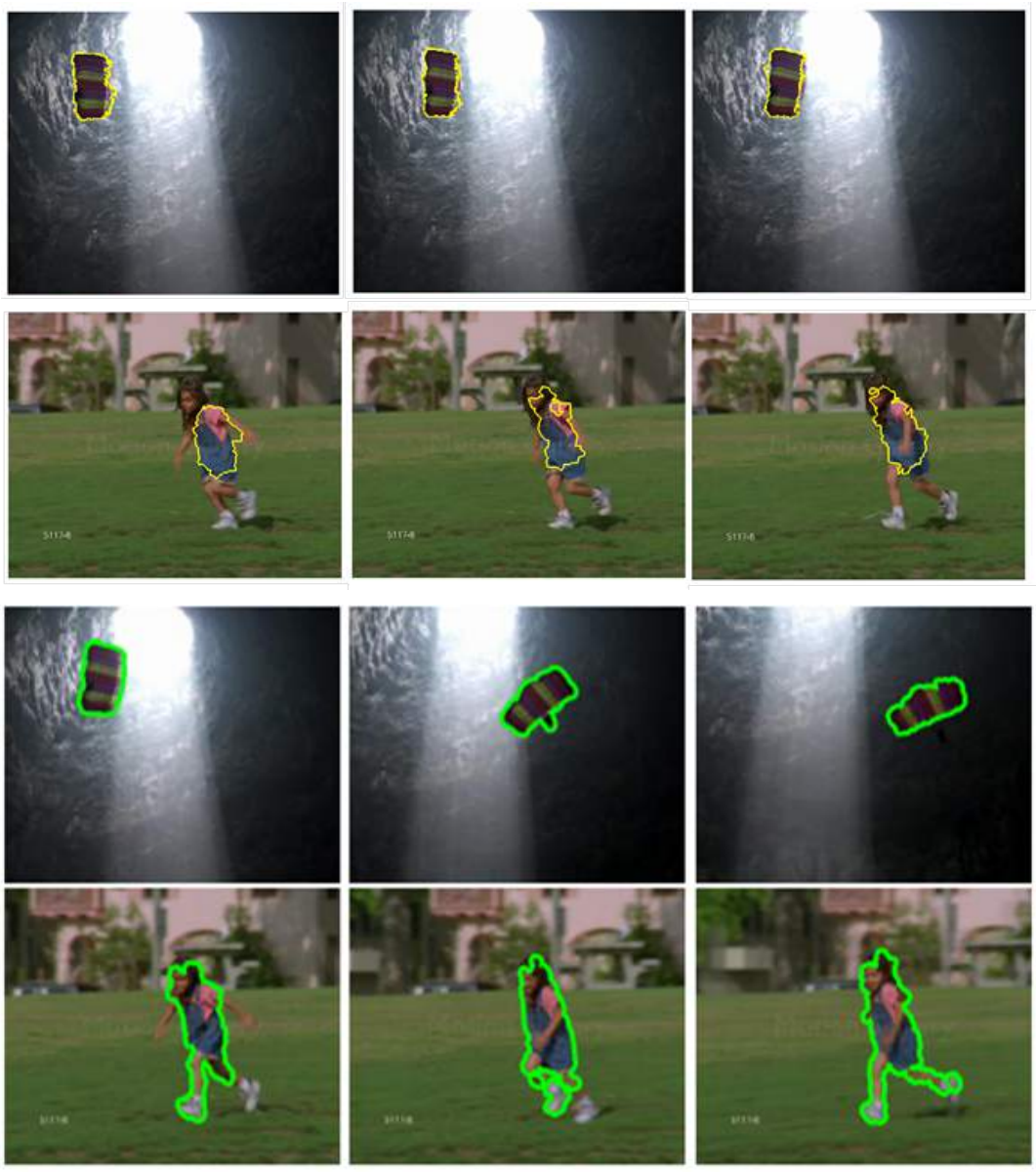

Figura 4.10: Comparación en Secuencia de imágenes de SegTrack. Fila 1 y 2, resultado del modelo propuesto. Fila 3 y 4, resultado del trabajo de [59]

Conclusión

Se desarrolló un sistema de seguimiento de objetos en video, bajo un marco científico, considerando una amplia revisión bibliográfica, de los últimos avances relacionados a seguimiento de objetos, diseñando e implementando la propuesta de investigación planteada. Obteniendo buenos resultados localizando espacial y temporalmente un único objeto por secuencia de video.

La investigación realizada permite concluir que las características de color combinadas con movimiento, permiten identificar y diferenciar entre un objeto y el fondo. Se puede decir que los descriptores utilizados, un modelo mixto gaussiano (GMM) para el modelo de color RGB, el histograma de flujo óptico para describir movimiento y los histogramas 3D de color para comparar segmentos, caracterizan de manera adecuada cada superpíxel y capturan la información necesaria para construir el grafo que entrega un etiquetado objeto/fondo para cada superpíxel. Se puede afirmar que el modelo planteado por medio de grafos, que se resuelve con un método de optimización basado en *Graph-Cut*, es un modelo sólido y que presenta buen funcionamiento en la tarea de seguir objetos articulados con precisión a nivel de contornos.

Se verificó el correcto funcionamiento del modelo propuesto en tres conjuntos de videos, Cars-MoSeg, SegTrack y MCCD. Se demostró que el algoritmo implementado presenta un buen desempeño, adaptándose a la forma del objeto de interés. Ante situaciones como cambios de iluminación, de puntos de vista, los movimientos de cámara, la presencia en la escena de elementos con apariencia similar al objeto y ante objetos articulados que van cambiando la forma de su contorno a través de la secuencia.

En los resultados obtenidos es posible encontrar un error asociado a la anotación manual o *groundtruth*, que puede tener pequeñas equivocaciones a nivel de píxeles.

Además, en las secuencias de video no se tiene una anotación por *frame*, por lo cual se tomaba un promedio del rendimiento del modelo, con sólo un 10 % de los *frames* evaluados. Estas condiciones, agregan errores en el desempeño cuantitativo del modelo.

Bibliografía

[1] "tv-smart tv — samsung." http://www.samsung.com/co/consumer/tv-audio-video/tv/smart-tv.

[2] "Kinect by Xbox." http://www.xbox.com/es-ES/kinect.

[3] A. Kovashka and K. Grauman, "Learning a hierarchy of discriminative space-time neighborhood features for human action recognition," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 2046–2053, IEEE, June 2010.

[4] Z. Kalal, K. Mikolajczyk, and J. Matas, "Forward-backward error: Automatic detection of tracking failures," in *2010 20th International Conference on Pattern Recognition*, pp. 2756–2759, IEEE, Aug. 2010.

[5] J. Prokaj and G. Medioni, "Using 3D scene structure to improve tracking," in *CVPR 2011*, pp. 1337–1344, IEEE, June 2011.

[6] R. R. Cabrera, T. Tuytelaars, and L. Van Gool, "Efficient multi-camera detection, tracking, and identification using a shared set of haar-features," in *CVPR 2011*, pp. 65–71, IEEE, June 2011.

[7] B. Benfold and I. Reid, "Stable multi-target tracking in real-time surveillance video," in *CVPR 2011*, pp. 3457–3464, IEEE, June 2011.

[8] V. Escorcia, M. A. Dávila, M. Golparvar-Fard, and J. C. Niebles, "Automated vision-based recognition of construction worker actions for building interior construction operations using rgbd cameras," in *Construction Research Congress 2012@ Construction Challenges in a Flat World*, pp. 879–888, ASCE, 2012.

[9] J. Xiao, H. Cheng, H. Sawhney, and F. Han, "Vehicle detection and tracking in wide field-of-view aerial video," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 679–684, IEEE, June 2010.

[10] M. Leotta and J. Mundy, "Predicting high resolution image edges with a generic, adaptive, 3-D vehicle model," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1311–1318, IEEE, June 2009.

[11] V. Frati and D. Prattichizzo, "Using kinect for hand tracking and rendering in wearable haptics," in *2011 IEEE World Haptics Conference*, pp. 317–321, IEEE, June 2011.

[12] S.-H. Lee and J.-S. Choi, "Real-time camera pose estimation based on planar object tracking for augmented reality environment," in *2012 IEEE International Conference on Consumer Electronics (ICCE)*, pp. 516–517, IEEE, Jan. 2012.

[13] S. Mulatto, A. Formaglio, M. Malvezzi, and D. Prattichizzo, "Using Postural Synergies to Animate a Low-dimensional Hand Avatar in Haptic Simulation," *IEEE Transactions on Haptics*, 2012.

[14] T. Nakamura, "Real-time 3-D object tracking using Kinect sensor," in *2011 IEEE International Conference on Robotics and Biomimetics*, pp. 784–788, IEEE, Dec. 2011.

[15] N. Owens, "Hawk-Eye tennis system," in *International Conference on Visual Information Engineering (VIE 2003). Ideas, Applications, Experience*, vol. 2003, pp. 182–185, IEE, 2003.

[16] B. Alexe, T. Deselaers, and V. Ferrari, "What is an object?," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 73–80, IEEE, June 2010.

[17] Z. Kalal, J. Matas, and K. Mikolajczyk, "Online learning of robust object detectors during unstable tracking," in *2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops*, pp. 1417–1424, IEEE, Sept. 2009.

[18] Z. Kalal, J. Matas, and K. Mikolajczyk, "P-N learning: Bootstrapping binary classifiers by structural constraints," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 49–56, IEEE, June 2010.

[19] "TLD — Zdenek Kalal." http://info.ee.surrey.ac.uk/Personal/Z.Kalal/tld.html.

[20] R. Szeliski, *Computer Vision: Algorithms and applications*. Springer, 2010.

[21] J. Shi and C. Tomasi, "Good features to track," in *Computer Vision and Pattern Recognition, 1994. Proceedings CVPR '94., 1994 IEEE Computer Society Conference on*, pp. 593–600, IEEE Comput. Soc. Press, Jun 1994.

[22] C. Yuan, X. Li, W. Hu, H. Ling, and S. Maybank, "3d r transform on spatio-temporal interest points for action recognition," in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pp. 724–730, June 2013.

[23] X. Yu, C. Xu, H. W. Leong, Q. Tian, Q. Tang, and K. W. Wan, "Trajectory-based ball detection and tracking with applications to semantic analysis of broadcast soccer video," in *Proceedings of the eleventh ACM international conference on Multimedia*, pp. 11–20, ACM, 2003.

[24] P. Ochs and T. Brox, "Object segmentation in video: A hierarchical variational approach for turning point trajectories into dense regions," in *2011 International Conference on Computer Vision*, pp. 1583–1590, IEEE, Nov. 2011.

[25] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1, pp. 886–893, IEEE, 2005.

[26] I. Endres and D. Hoiem, "Category independent object proposals," *Computer Vision–ECCV 2010*, pp. 575–588, 2010.

[27] B. Alexe, T. Deselaers, and V. Ferrari, "Measuring the objectness of image windows.," *IEEE transactions on pattern analysis and machine intelligence*, vol. 34, pp. 2189–202, Nov. 2012.

[28] K. E. A. van de Sande, J. R. R. Uijlings, T. Gevers, and A. W. M. Smeulders, "Segmentation as selective search for object recognition," in *2011 International Conference on Computer Vision*, pp. 1879–1886, IEEE, Nov. 2011.

[29] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient Graph-Based Image Segmentation," *International Journal of Computer Vision*, vol. 59, pp. 167–181, Sept. 2004.

[30] J. C. Niebles, B. Han, A. Ferencz, and L. Fei-fei, "Extracting moving people from internet videos," *IN ECCV*, 2008.

[31] J. C. Niebles, B. Han, and L. Fei-Fei, "Efficient extraction of human motion volumes by tracking," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 655–662, IEEE, June 2010.

[32] C. Xu and J. J. Corso, "Evaluation of super-voxel methods for early video processing," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1202–1209, IEEE, June 2012.

[33] O. Veksler, Y. Boykov, and P. Mehrani, "Superpixels and Supervoxels in an Energy Optimization Framework," in *European Conference on Computer Vision*, pp. 211–224, 2010.

[34] Y. J. Lee, J. Kim, and K. Grauman, "Key-segments for video object segmentation," in *2011 International Conference on Computer Vision*, pp. 1995–2002, IEEE, Nov. 2011.

[35] B. Zhang, H. Zhao, and X. Cao, "Video object segmentation with shortest path," in *Proceedings of the 20th ACM international conference on Multimedia - MM '12*, (New York, New York, USA), p. 801, ACM Press, 2012.

[36] Z. Tian, J. Xue, X. Lan, C. Li, and N. Zheng, "Object segmentation and key-pose based summarization for motion video," *Multimedia Tools and Applications*, May 2013.

[37] G. Sharir and T. Tuytelaars, "Video object proposals," in *2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pp. 9–14, IEEE, June 2012.

[38] J. Lezama, K. Alahari, J. Sivic, and I. Laptev, "Track to the future: Spatio-temporal video segmentation with long-range motion cues," in *CVPR 2011*, pp. 3369–3376, IEEE, June 2011.

[39] H. Y. LeCun, S., Chopra, R., "A Tutorial on Energy-Based Learning," in *Predicting Structured Data*, MIT Press, 2006.

[40] G. Bakir, T. Hofman, B. Scholkopf, A. Smola, and B. Taskar, *Predicting Structured Data*. MIT Press, 2007.

[41] B. Fulkerson, A. Vedaldi, and S. Soatto, "Class segmentation and object localization with superpixel neighborhoods," in *2009 IEEE 12th International Conference on Computer Vision*, pp. 670–677, IEEE, Sept. 2009.

[42] N. Xu, R. Bansal, and N. Ahuja, "Object segmentation using graph cuts based active contours," in *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, vol. 2, pp. II–46–53 vol.2, 2003.

[43] B. Yao and L. Fei-Fei, "Action recognition with exemplar based 2.5 d graph matching," in *Computer Vision–ECCV 2012*, pp. 173–186, Springer, 2012.

[44] M.-H. Y. Yi Wu, Jongwoo Lim, "Online Object Tracking: A Benchmark," in *Computer Vision Pattern Recognition*, IEEE, 2013.

[45] S. Stalder, H. Grabner, and L. Van Gool, "Dynamic objectness for adaptive tracking," pp. 43–56, 2012.

[46] S. S. Tabatabaei, M. Coates, and M. Rabbat, "GANC: Greedy agglomerative normalized cut for graph clustering," *Pattern Recognition*, vol. 45, pp. 831–843, Feb. 2012.

[47] "Tracker Benchmark v1.0 — Visual Tracker Benchmark." http://cvlab.hanyang.ac.kr/wordpress/?page_id=14.

[48] "Berkeley Motion Segmentation Dataset." http://lmb.informatik.uni-freiburg.de/resources/datasets/moseg.en.html.

[49] D. Tsai, M. Flagg, A. Nakazawa, and J. M. Rehg, "Motion Coherent Tracking Using Multi-label MRF Optimization," *International Journal of Computer Vision*, vol. 100, pp. 190–202, Dec. 2011.

[50] D. Tsai, M. Flagg, A. Nakazawa, and J. M. Rehg, "SegTrack." http://cpl.cc.gatech.edu/projects/SegTrack/.

[51] "Piotr's Image Video Matlab Toolbox." http://vision.ucsd.edu/~pdollar/toolbox/doc/.

[52] "VLFeat open source library." http://www.vlfeat.org/.

[53] P. Chockalingam, N. Pradeep, and S. Birchfield, "Adaptive fragments-based tracking of non-rigid objects using level sets," in *Computer Vision, 2009 IEEE 12th International Conference on*, pp. 1530–1537, IEEE, 2009.

[54] T. Ma and L. J. Latecki, "Maximum weight cliques with mutex constraints for video object segmentation," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pp. 670–677, IEEE, 2012.

[55] D. Zhang, O. Javed, and M. Shah, "Video Object Segmentation through Spatially Accurate and Temporally Dense Extraction of Primary Object Regions," in *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 628–635, IEEE, June 2013.

[56] B. Thomas and M. Jitendra, "Object Segmentation by Long Term Analysis of Point Trajectories," *In Proc. European Conference on Computer Vision*, pp. 282–295, 2010.

[57] P. Ochs and T. Brox, "Higher order motion models and spectral clustering," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pp. 614–621, IEEE, 2012.

[58] O. Barnich and M. Van Droogenbroeck, "Vibe: A universal background subtraction algorithm for video sequences," *Image Processing, IEEE Transactions on*, vol. 20, no. 6, pp. 1709–1724, 2011.

[59] A. Papazoglou and V. Ferrari, "Fast Object Segmentation in Unconstrained Video," in *2013 IEEE International Conference on Computer Vision*, pp. 1777–1784, IEEE, Dec. 2013.